Quantification of Prediction Uncertainty for Principal Components Regression and Partial Least Squares Regression

by

Ying Zhang

A Thesis Submitted for the Degree of

Doctor of Philosophy

in the

Faculty of Mathematical & Physical Sciences

Department of Statical Science

University College London

Declaration of Authorship

I, , confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Abstract

Principal components regression (PCR) and partial least squares regression (PLS) are widely used in multivariate calibration in the fields of chemometrics, econometrics, social science and so forth, serving as alternative solutions to the problems which arise in ordinary least squares regression when explanatory variables are either collinear, or there are hundreds of explanatory variables with a relatively small sample size. Both PCR and PLS tackle the problems by constructing lower dimensional factors based on the explanatory variables.

The extra step of factor construction makes the standard prediction uncertainty theory of ordinary least squares regression not directly applicable to the two reduced dimension methods. In the thesis, we start by reviewing the ordinary least squares regression prediction uncertainty theory, and then investigate how the theory performs when it extends to PCR and PLS, aiming at potentially better approaches.

The first main contribution of the thesis is to clarify the quantification of prediction uncertainty for PLS. We rephrase existing methods with consistent mathematical notations in the hope of giving a clear guidance to practitioners.

The second main contribution is to develop a new linearisation method for PLS. After establishing the theory, simulation and real data studies have been employed to understand and compare the new method with several commonly used methods.

From the studies of simulations and a real dataset, we investigate the properties of simple approaches based on the theory of ordinary least squares theory, the approaches using resampling of data, and the local linearisation approaches including a classical and our improved new methods. It is advisable to use the ordinary least squares type prediction variance with the estimated regression error variance from the tuning set in both PCR and PLS in practice.

Acknowledgements

First and foremost, I would like to thank my primary supervisor, Professor Thomas Fearn. I am greatly indebted to him, for sharing his intellect with me, his highly professional guidance, and his patience to read my reports and drafts. I feel very fortunate to have had the opportunity to study with him. I would like to thank my parents, Qingping Zhang and Pu Wang, for their generous support and encouragement. They are my best friends in life. Without them, I would not have been able to pursue my interests. I am very thankful to my secondary supervisor, Dr. Jinghao Xue, for his kind suggestions about my upgrade report and my PhD research life.

I would like to express sincere gratitude to the department of Statistical Science and its staff. I am especially grateful to my MSc advisor, Dr. Rex Galbraith, for opening the door of Statistics for me. I greatly appreciate the guidance from my MSc research project supervisors, Professor Richard Chandler and Dr. Afzal Siddiqui. I would like to thank Ms. Marion Ware and Dr. Ying Zhu for their help and advice on my applications to the MSc and PhD programs.

I would like to thank Goodenough College for providing me a comfortable home in London. I would also like to thank my friends, for accompanying me during the PhD study.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The use of multivariate calibration in chemistry is most strongly associated with quantitative near infrared (NIR) spectroscopy, although it is increasingly being used in other applications. The idea is to produce predictions of sample composition from a multivariate measurement such as a near infrared spectrum. This involves fitting a prediction equation, in the simplest case a linear one, to data on a training or calibration set of samples for which we know both spectra and composition. The signals at different wavelengths are taken as explanatory variables, while the chemical composition is the response variable (or variables). When the number of explanatory variables is large (a spectrum is typically measured at 1000 wavelengths or so) standard methods such a multiple linear regression break down and in what is often called chemometrics a number of alternatives have been invented to cope with this. The two best-known approaches are principal components regression (PCR) and partial least squares regression (PLS). Both work by constructing new variables (factors) that contain most of the information on the spectral data in a much smaller number of variables and fitting a regression equation using these new variables. Principal components regression constructs its factors via a principal component analysis of the spectral data. Partial least squares regression works in a similar way, but the construction of the factors involves both the explanatory variables and the response variable or variables. These methods have been used successfully for some time now, and many of their proper-

ties are fairly well understood. One area that is still not well understood however is how to quantify prediction uncertainty from the calibration equations. Although it is clear how to do this in the case of a multiple linear regression, the extra step of factor construction in principal components regression and partial least squares regression means that this standard theory is not applicable directly to these cases. This is especially true for partial least squares regression where the response variable is involved in the construction of the factors and thus contributes noise to them.

1.1 Multiple Linear Regression and Prediction Uncertainty

A multiple linear regression model of a calibration set for a single response variable can be written as

$$\dot{\mathbf{y}}_c = \beta_0 + \dot{\mathbf{X}}_c\boldsymbol{\beta} + \boldsymbol{\xi}_c, \tag{1.1}$$

where $\dot{\mathbf{y}}_c$ $(n \times 1)$ is the response variable in the calibration set, $\dot{\mathbf{X}}_c$ $(n \times k)$ are explanatory variables, β_0 is an intercept, $\boldsymbol{\beta}$ $(k \times 1)$ are regression coefficients, and ξ_c $(n \times 1)$ is an error term that is independently and identically normally distributed with mean 0 and variance σ^2_ξ. The multiple linear regression model is often written in terms of centred explanatory variables for computational convenience, thus

$$\dot{\mathbf{y}}_c = \alpha + \mathbf{X}_c\boldsymbol{\beta} + \boldsymbol{\xi}_c, \tag{1.2}$$

where the centred calibration explanatory variables $\mathbf{X}_c = \dot{\mathbf{X}}_c - \mathbf{1}\bar{\dot{\mathbf{x}}}$, where $\bar{\dot{\mathbf{x}}}$ $(1 \times k)$ is the mean of the explanatory variables, and $\mathbf{1}$ is an $n \times 1$ vector, all of whose elements are ones. The scalar α denotes the intercept in the case of centred explanatory variables. In the thesis, we use the dot on top as a notation to denote non-centred observations, whilst the notations without the dot are either centred values or quantities derived from the centred values.

After regression coefficients have been estimated, as $\hat{\alpha}$ and $\hat{\boldsymbol{\beta}}$, a predicted value

for a new observation with explanatory variable $\dot{\mathbf{x}}_p$ $(1 \times k)$ can be calculated as

$$\begin{aligned} \hat{\dot{y}}_p &= \hat{\beta}_0 + \dot{\mathbf{x}}_p \hat{\boldsymbol{\beta}} \qquad \text{where } \hat{\beta}_0 = \bar{\dot{y}} - \bar{\dot{\mathbf{x}}}\hat{\boldsymbol{\beta}} \qquad (1.3) \\ &= \hat{\alpha} + \mathbf{x}_p \hat{\boldsymbol{\beta}} \qquad \text{where } \hat{\alpha} = \bar{\dot{y}}. \qquad (1.4) \end{aligned}$$

where $\mathbf{x}_p$ denotes the centred predictor, $\mathbf{x}_p = \dot{\mathbf{x}}_p - \bar{\dot{\mathbf{x}}}$. Note that the centring is done with the calibration set mean. For an ordinary least squares regression $\hat{\dot{y}}_p$ is unbiased for $\dot{y}_p$, and

$$\text{Var}\,(\hat{\dot{y}}_p) = \sigma_\xi^2 (\frac{1}{n} + h), \qquad (1.5)$$

where the leverage h is defined as $h = \mathbf{x}_p(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_p'$. Another useful measure of prediction uncertainty is the variance of the difference between the observed value and the predicted value

$$\begin{aligned} \text{Var}\,(\dot{y}_p - \hat{\dot{y}}_p) &= \text{E}\,(\dot{y}_p - \hat{\dot{y}}_p)^2 \\ &= \sigma_\xi^2 (\frac{1}{n} + h + 1). \qquad (1.6) \end{aligned}$$

Depending on the context, either Equation (1.5) or (1.6) might be regarded as quantifying prediction uncertainty. We will use Equation (1.6) more often, since it relates more closely to observed quantities. To use the prediction uncertainty formulae in Equations (1.5) and (1.6), we need an estimate of σ_ξ^2. This can be obtained from the residual variance in the calibration set, called mean squared error of calibration (MSEC) or squared standard error of calibration (SEC^2) in the chemometrics literature,

$$\text{MSEC} = \frac{1}{n-k-1} \sum_{j=1}^{n} (\dot{y}_{c_j} - \hat{\alpha} - \mathbf{x}_{c_j}\hat{\boldsymbol{\beta}})^2. \qquad (1.7)$$

where $\mathbf{x}_{c_j} = \dot{\mathbf{x}}_{c_j} - \bar{\dot{\mathbf{x}}}$ denotes the j-th centred explanatory variable row vector in the calibration set.

Ordinary least squares regression is the simplest regression method, and its prediction uncertainty theory is well established. Equations (1.5) and (1.6) are exact if the model is correct, and the MSEC of multiple linear regression is an unbiased estimator of the regression error variance σ_ξ^2. There is not a well-developed theory for principal components regression, and especially for partial least squares

regression. In the case of partial least squares regression, for example, it is not even clear what should be the divisor in Equation (1.7). In the absence of theory, one commonly adopted empirical approach to estimating prediction uncertainty is to use a second set of samples, which we will call a tuning set, as follows.

- Fit the prediction equation using the calibration set.
- Use the tuning set, a separate set of data from the calibration set, obtained under the same conditions, to estimate the root mean squared error of prediction (RMSEP).
- Use this RMSEP as the standard deviation attaching to any future prediction.

The use of RMSEP derived in this way is a simple approach to estimating prediction uncertainty, which works regardless of the algorithm that produces the prediction equation. Assume the calibration set has n observations $\{\dot{\mathbf{y}}_c, \dot{\mathbf{X}}_c\}$, and the tuning set has n_t observations $\{\dot{\mathbf{y}}_t, \dot{\mathbf{X}}_t\}$. The estimates of regression coefficients $\hat{\alpha}$ and $\hat{\boldsymbol{\beta}}$ are obtained from the calibration set. Predictions are calculated for the tuning set, then

$$\text{RMSEP} = \sqrt{\frac{1}{n_t}\sum_{j=1}^{n_t}(\dot{y}_{t_j} - \hat{\dot{y}}_{t_j})^2} = \sqrt{\frac{1}{n_t}\sum_{j=1}^{n_t}\{\dot{y}_{t_j} - \hat{\alpha} - \mathbf{x}_{t_j}\hat{\boldsymbol{\beta}}\}^2}, \tag{1.8}$$

where $\mathbf{x}_{t_j}$ is the centred explanatory variables of the j-th observation in the tuning set, $\mathbf{x}_{t_j} = \dot{\mathbf{x}}_{t_j} - \bar{\dot{\mathbf{x}}}$. Note that the centring is once again done with the calibration set mean. The limitation of this approach is of course that it attaches the same variance to all predictions. The basic challenge for this thesis is to try to improve on this, taking into account the value of $\mathbf{x}_p$ when quantifying the uncertainty in the predictions.

In the case where the prediction equation has been estimated by multiple linear regression we could use RMSEP from the tuning set to estimate the regression error variance σ_ξ^2 in Equation (1.6). Mean squared error of prediction (MSEP), that is

the squared RMSEP, can be used as follows:

$$\hat{\sigma}_\xi^2 = \frac{\text{MSEP}}{\frac{1}{n_t} + \frac{1}{n_t}\sum_{j=1}^{l} h_{t_j} + 1}, \tag{1.9}$$

where h_{t_j} is the leverage for the j-th observation in the tuning set, $h_{t_j} = \mathbf{x}_{t_j}(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_{t_j}'$. We could use Equation (1.9) instead of Equation (1.7) to substitute for σ_ξ^2 in Equation (1.6) as the variance of future predictions. This rather complicated way of proceeding is unnecessary in the case of multiple linear regression, but will be useful later when dealing with approximate prediction uncertainty formulae for principal components regression and partial least squares regression that need an estimate of a regression error variance. Using an approach analogous to the one above will at least ensure that the average prediction uncertainty estimate is correct.

1.2 Principal Components Regression (PCR) and Partial Least Squares Regression (PLS)

Partial least squares regression was introduced into econometrics by the Swedish statistician Herman Wold. His son, Svante Wold, and Harald Martens pioneered the development of partial least squares methods in chemometrics from the late 1970s (Wold et al. (1983)). Since then partial least squares regression has been widely used in the field of chemometrics and in application areas such as food research, bioinformatics and medicine. Partial least squares methods are generally presented in terms of algorithms, of which there are many. For example Andersson (2009) compares the numerical stability of nine algorithms. Wold (1966) proposes the nonlinear iterative partial least squares method (NIPALS), which was first called the non-linear estimation by iterative least square procedures (NILES). It is also called the orthogonal scores algorithm. Another standard algorithm, SIMPLS, was presented by De Jong (1993) as a "straightforward implementation of a statistically inspired modification of the PLS method according to a simple concept".

There are several good introductory works on partial least squares regression,

for example Martens and Næs (1991), Geladi and Kowalski (1986), Wold et al. (2001), Rosipal and Krämer (2006). As the applications of partial least squares regression increased, its mathematical and statistical properties became of interest. Höskuldsson (1988), Helland (1988), Helland (1990), and Stoica and Söderström (1998) all study these properties. Partial least squares regression has been connected to other regression methods, for example principal components regression and ridge regression. Several of these methods can be unified under an approach called continuum regression (Stone and Brooks (1990), Frank and Friedman (1993), Dunn III et al. (1989), Björkström and Sundberg (1996)).

Principal components regression and partial least squares regression are typically used when $\mathbf{X}_c'\mathbf{X}_c$ is either singular, because there are more explanatory variables than response variables, or ill-conditioned, because the explanatory variables are strongly correlated. Both methods construct new explanatory variables or factors as linear combinations of the original explanatory variables. Principal components regression reduces the original explanatory variables to a smaller number of so-called principal components (PCs), which capture as much as possible of the variability in these explanatory variables. The PCs correspond to the eigenvectors of $\mathbf{X}_c'\mathbf{X}_c$ that have the largest eigenvalues. Partial least squares regression constructs its factors to maximise the covariance between the constructed factors and the response variable. In either case ordinary least squares estimation is carried out using the scores of these factors as predictors.

1.3 The Problem of Prediction Uncertainty

If we assume that the model generating the observed calibration data is a linear regression, it can be written as

$$\dot{\mathbf{y}}_c = \beta_0 + \dot{\mathbf{X}}_c\boldsymbol{\beta} + \boldsymbol{\epsilon}_c, \tag{1.10}$$

where the response variable $\dot{\mathbf{y}}_c$, the explanatory variables $\dot{\mathbf{X}}_c$, and the regression coefficients β_0 and $\boldsymbol{\beta}$ are defined as the same as those in the multiple linear regression model, Equation (1.1). One change in notation from Equation (1.1) is

that $\boldsymbol{\epsilon}_c$ now denotes the error term instead of $\boldsymbol{\xi}_c$. This has been done because we need to distinguish carefully between the error term in the equation generating the data, and that in the fitted principal components regression or partial least squares regression equation. The error term $\boldsymbol{\xi}_c$ has been reserved for use in the second of these equations.

The general form of the bilinear model for centred explanatory variables $\mathbf{X}_c$ ($n \times k$) and a centred single response variable $\mathbf{y}_c$ ($n \times 1$), used in principal components regression and partial least squares regression can be expressed as follows:

$$\begin{aligned} \mathbf{V}_a &= f(\mathbf{X}_c, \mathbf{y}_c), \\ \mathbf{T} &= \mathbf{X}_c \mathbf{V}_a, \\ \mathbf{X}_c &= \mathbf{T}\mathbf{P}' + \mathbf{E}, \\ \mathbf{y}_c &= \mathbf{T}\mathbf{q}' + \mathbf{f}. \end{aligned} \tag{1.11}$$

The weight matrix $\mathbf{V}_a$ ($k \times a$) is a function of the centred data matrix $\mathbf{X}_c$ and the centred response vector $\mathbf{y}_c$. The explanatory variables and the response variable are connected by the latent variables $\mathbf{T}$ ($n \times a$), called scores, with loadings $\mathbf{P}$ ($k \times a$) and $\mathbf{q}$ ($1 \times a$). $\mathbf{E}$ ($n \times k$) is the residual matrix from the regression of the centred explanatory variables on the scores. $\mathbf{f}$ ($n \times 1$) denotes the regression error from the regression of $\mathbf{y}_c$ on the same scores, and corresponds to the regression error $\boldsymbol{\xi}_c$ in the multiple linear regression model, Equation (1.1) or (1.2). In this reduced dimension model, $\mathbf{f}$ includes both the random variation about the regression of Equation (1.10) and the bias due to the dimension reduction from $\mathbf{X}_c$ to $\mathbf{T}$. The bias will not be linearly dependent on $\mathbf{T}$, but may be dependent on the part of $\mathbf{X}_c$ orthogonal to $\mathbf{T}$. In order to carry out the ordinary least squares regression in the last step, we treat $\mathbf{f}$ as random, and assume it has a normal distribution with mean zero and variance σ_ξ^2.

In principal components regression, the computation of principal components does not involve the response variable, although the response variable is often used in a cross-validation to choose the number of factors, and thus decide the dimension of the weight matrix. The weight matrix $\mathbf{V}_a$ for principal components

regression is a truncated version of $\mathbf{V}$, which consists of the eigenvectors of $\mathbf{X}_c'\mathbf{X}_c$, and because of the properties of the eigenvectors $\mathbf{P} = \mathbf{V}_a$. Ordinary least squares estimation is employed to estimate the loadings, so that

$$\hat{\mathbf{q}}' = (\mathbf{T}'\mathbf{T})^{-1}\mathbf{T}'\mathbf{y}_c.$$

The scores of the predictors for a new sample are denoted as $\mathbf{t}_p = \mathbf{x}_p\hat{\mathbf{P}}$.

In partial least squares regression, the loadings $\hat{\mathbf{q}}$ and the predictor scores $\mathbf{t}_p$ are calculated sequentially, and their mathematical form varies according to different algorithms, as will be described in Section 4.1.

For both principal components regression and partial least squares regression, the prediction can be written as,

$$\hat{\bar{y}}_p = \bar{y} + \mathbf{t}_p\hat{\mathbf{q}}'. \tag{1.12}$$

The last step of the bilinear model, Equation (1.11), corresponds to the multiple linear regression model Equation (1.2), where the response variable is a linear function of centred explanatory variables. Similarly Equation (1.12), like Equation (1.4), gives the linear relationship between the observed prediction $\hat{\bar{y}}_p$ and the predictors $\mathbf{t}_p$. From Equation (1.12) we can see the prediction uncertainty can be decomposed into three parts with respect to the uncertainty in $\bar{y}$, $\mathbf{t}_p$ and $\hat{\mathbf{q}}$ separately. The variance of $\bar{y}$ equals to $\frac{\sigma_\xi^2}{n}$, where σ_ξ^2 is the variance in $\mathbf{y}_c$ not explained by the regression on the scores. The variation in $\mathbf{t}_p$ comes from constructing factors from the original explanatory variables, while the variation in $\hat{\mathbf{q}}$ comes from the ordinary least squares estimation. For $\hat{\mathbf{q}}$, we can use the ordinary least squares prediction variance, but the variation in $\mathbf{t}_p$ depends on the method used to construct the factors.

In principal components regression, $\mathbf{t}_p$ appears only to depend on $\mathbf{X}_c$, although in practice the number of factors relies on $\mathbf{y}_c$ when we use cross-validation to choose this number. However, it could be argued that regarding $\mathbf{t}_p$ as fixed in this case is reasonable.

In contrast to the case of principal components regression where the weight matrix $\mathbf{V}_a$ consists of eigenvectors of $\mathbf{X}_c'\mathbf{X}_c$, the weight matrix $\mathbf{V}_a$ in partial least

squares regression depends on both $\mathbf{X}_c$ and $\mathbf{y}_c$ since the factors maximise the covariance between the response and constructed explanatory variables. Thus the scores $\mathbf{T}$ and $\mathbf{t}_p$ depend on both $\mathbf{y}_c$ and $\mathbf{X}_c$. Ignoring this would underestimate the prediction uncertainty.

In the thesis, we will review the ordinary least squares regression prediction theory in Chapter 2. Simulation studies will be used to reproduce the theoretical results, which lays a foundation for the study of prediction uncertainty in principal components regression and partial least squares regression. In Chapter 3, we will study the basic principal components regression theory, and its empirical and theoretical prediction uncertainty measurements, looking for alternative approaches to estimating principal components prediction uncertainty. For partial least squares regression, we will study various partial least squares algorithms, and summarise related works on prediction uncertainty in Chapter 4. In Chapter 5, we will present a new linearisation method estimating partial least squares prediction mean squared error, and compare it with other standard approaches using a simulation study and real data analysis. We hope the thesis will be helpful for understanding different approaches to quantifying prediction uncertainty in principal components regression and partial least squares regression, and provide a clear guidance on how to attach appropriate uncertainty to future predictions.

For ordinary least squares regression, prediction variance is equivalent to prediction mean squared error as it is an unbiased regression method, i.e. $\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \mathrm{Var}\left(\dot{y}_p - \hat{\dot{y}}_p\right)$. Principal components regression and partial least squares regression are biased regression methods, so $\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \mathrm{Var}\left(\dot{y}_p - \hat{\dot{y}}_p\right) + \mathrm{E}\left(\mathrm{bias}^2\right)$, where the bias is caused by the construction of reduced dimensional latent factors from explanatory variables. Most works in chemometrics use prediction variance as the estimate of prediction uncertainty, which actually should be squared prediction error because these studies do not concern the bias. Therefore, in this thesis, we study prediction mean squared error as an estimate of prediction uncertainty for principal components regression and partial least squares regression.

Prediction versus extrapolation arises in Copas (1983) approach to biased esti-

mation and does not necessarily involve leverage, it depends on directions of large effects and whether they are ill estimated. In ordinary least squares regression prediction mean squared error has a linear relationship with the leverage, so it is consistent if we study the relationship between prediction mean squared error and leverage in principal components regression and partial least squares regression. The leverage allows us to attach an prediction uncertainty measure to a particular prediction.

1.4 Notation

For convenience we collect together here some of the notations that will be used throughout the thesis. In general, bold capitals will be used for matrices, bold lower case symbols for vectors, and italics for scalars.

$\dot{\mathbf{X}}_c$	Explanatory variables in a calibration set
$\dot{\mathbf{y}}_c$	Single response variable in a calibration set
$\dot{\mathbf{X}}_t$	Explanatory variables in a tuning set
$\dot{\mathbf{y}}_t$	Single response variable in a tuning set
$\dot{\mathbf{x}}_p$	A row vector of predictors for a prediction sample
$\dot{y}_p$	An observed value for a prediction sample
$\bar{\dot{\mathbf{x}}}$	Mean of explanatory variables in the calibration set, a row vector
$\bar{\dot{y}}$	Mean of the single response variable in the calibration set
$\mathbf{X}_c$	Centred explanatory variables in the calibration set
$\mathbf{y}_c$	Centred single response variables in the calibration set
$\mathbf{x}_t$	Centred predictors $\mathbf{x}_t = \dot{\mathbf{x}}_t - \bar{\mathbf{x}}$ for a sample in the tuning set
$\mathbf{x}_p$	Centred predictors $\mathbf{x}_p = \dot{\mathbf{x}}_p - \bar{\mathbf{x}}$ for a prediction sample
$\mathbf{T}$	Scores of the factors in the calibration set
$\mathbf{t}_p$	Scores of the factors for a prediction sample
k	Number of explanatory variables
a	Number of factors chosen in PCR and PLS
h	OLS leverage $h = \mathbf{x}_p(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_p'$ PCR & PLS leverage $h = \mathbf{t}_p(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_p'$
hb	PLS leverage calculated from bootstrapping by residual
$Hden$	PLS distance measure defined in Denham's linearisation method
H	PLS distance measure defined by the new local linearisation method
Hb	PLS distance measure defined by the new local linearisation embedded with bootstrapping method
∂	Partial differentiation
vecut	An operator that gives a column vector whose elements are taken in order along rows including the diagonal elements from the upper triangular part of a symmetric matrix
diag	An operator that extracts the diagonal terms from a symmetric matrix as a column vector

Chapter 2

Ordinary Least Squares Regression Prediction Uncertainty

Ordinary least squares regression is used to estimate the loadings for the constructed factors in the last step of principal components regression and partial least squares regression. In Section 2.1 we review the ordinary least squares regression theory, and use simulation studies in Section 2.2 to reproduce the theoretical results, which paves the way for studying principal components regression in Chapter 3 and partial least squares regression in Chapter 5.

- Section 2.2.1 gives the simulation methodology. It runs simulations to verify the ordinary least squares prediction variance formula. The predictors are simulated independently from a common normal distribution.
- Section 2.2.2 shows that it is inappropriate to use an artificial setup: noise free prediction samples with uniformly distributed leverage, to study ordinary least squares prediction variance by simulation. Meanwhile, the need to simulate repeatedly at least the response variable in the calibration set is noted, and the implications for assessing the performance of any variance formula using a fixed 'real' calibration set are discussed.

- We examine the use of a tuning set (Section 2.2.3) and cross-validation (Section 2.2.4) to estimate empirically the approximate prediction variance for a fixed calibration set. The tuning set and the cross-validation can also provide simple empirical estimates for the prediction uncertainty as well as estimated regression error variances for the ordinary least squares prediction variance formula. Trying to find a way to round the problem to assess the performance of a prediction uncertainty formula using a real dataset, we study random data splitting in Section 2.2.5.

2.1 Ordinary Least Squares Regression Theory

We use the regression model $\dot{\mathbf{y}}_c = \alpha + \mathbf{X}_c\boldsymbol{\beta} + \boldsymbol{\xi}_c$ (Equation (1.2)), which for centred $\mathbf{y}_c$ as well as $\mathbf{X}_c$ becomes $\mathbf{y}_c = \mathbf{X}_c\boldsymbol{\beta} + \boldsymbol{\xi}_c$. Ordinary least squares regression minimises the residual sum of squares $\text{RSS} = (\mathbf{y}_c - \mathbf{X}_c\boldsymbol{\beta})'(\mathbf{y}_c - \mathbf{X}_c\boldsymbol{\beta})$, to find estimated regression coefficients. Differentiating,

$$\frac{\partial \text{RSS}}{\partial \boldsymbol{\beta}} = -2\mathbf{X}_c'\mathbf{y}_c + 2\mathbf{X}_c'\mathbf{X}_c\boldsymbol{\beta},$$

and setting this to 0 leads to the regression coefficient estimates,

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{X}_c'\mathbf{y}_c, \tag{2.1}$$

and $\hat{\alpha} = \bar{y}$. Replacing $\mathbf{y}_c$ in Equation (2.1), $\hat{\boldsymbol{\beta}} = (\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{X}_c'(\mathbf{X}_c\boldsymbol{\beta} + \boldsymbol{\xi}_c) = \boldsymbol{\beta} + (\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{X}_c'\boldsymbol{\xi}_c$, which gives $\hat{\boldsymbol{\beta}} - \boldsymbol{\beta} = (\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{X}_c'\boldsymbol{\xi}_c$. Thus the variance of the estimated regression coefficients is

$$\begin{aligned}
\text{Var}\,(\hat{\boldsymbol{\beta}}) &= \text{E}\,\{(\hat{\boldsymbol{\beta}} - \text{E}\,(\hat{\boldsymbol{\beta}}))(\hat{\boldsymbol{\beta}} - \text{E}\,(\hat{\boldsymbol{\beta}}))'\} = \text{E}\,\{(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'\} \\
&= (\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{X}_c'\text{Var}\,(\boldsymbol{\xi}_c\boldsymbol{\xi}_c')\mathbf{X}_c(\mathbf{X}_c'\mathbf{X}_c)^{-1} \\
&= \sigma_\xi^2(\mathbf{X}_c'\mathbf{X}_c)^{-1}.
\end{aligned} \tag{2.2}$$

The prediction variance

$$\begin{aligned} \mathrm{Var}\,(\hat{\dot{y}}_p) &= \mathrm{Var}\,(\hat{\alpha} + \mathbf{x}_p\hat{\boldsymbol{\beta}}) \\ &= \mathrm{Var}\,(\bar{\dot{y}}) + \mathbf{x}_p\mathrm{Var}\,(\hat{\boldsymbol{\beta}})\mathbf{x}_p' \\ &= \frac{\sigma_\xi^2}{n} + \sigma_\xi^2\mathbf{x}_p(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_p' \\ &= \sigma_\xi^2(\frac{1}{n} + h) \end{aligned} \tag{2.3}$$

and,

$$\mathrm{Var}\,(\dot{y}_p - \hat{\dot{y}}_p) = \sigma_\xi^2(1 + \frac{1}{n} + h), \tag{2.4}$$

where $h = \mathbf{x}_p(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_p'$, so that prediction variance and leverage h have a linear relationship with a slope of σ_ξ^2, and an intercept of $\sigma_\xi^2(1+\frac{1}{n})$. An unbiased estimator of the regression error variance is $\hat{\sigma}_\xi^2 = \frac{1}{n-k-1}\sum_{j=1}^{n}(\dot{y}_c - \hat{\dot{y}}_c)^2$.

2.2 Ordinary Least Squares Regression Simulation Study

2.2.1 Methodology

To understand how the prediction error associates with the leverage in principal components regression and partial least squares regression, we first try to reproduce the known relationship for ordinary least squares regression, in the expectation that the simple ordinary least squares regression simulation will give us a guidance to design simulations for principal components regression and partial least squares regression.

The linear models for a single response variable can be expressed as

$$\begin{aligned} \dot{\mathbf{y}}_c &= \beta_0 + \dot{\mathbf{X}}_c\boldsymbol{\beta} + \boldsymbol{\xi}_c, \\ \dot{\mathbf{y}}_p &= \beta_0 + \dot{\mathbf{X}}_p\boldsymbol{\beta} + \boldsymbol{\xi}_p, \end{aligned}$$

where the subscript $_c$ denotes the variables in the calibration set, and the subscript $_p$ denotes the variables for the prediction samples. Let j index the observations:

in the calibration set $j = 1, \ldots, n$, and in the prediction set $j = 1, \ldots, n_p$. A number k of explanatory variables is of interest. There will be N replicates in the simulation. The simulation studies are designed taking into account the following

- As the prediction samples are similar to those in the calibration set, the prediction set is generated to have the same average leverage as the calibration set.

- A noise term is not included in the simulation of the response variable for prediction samples, because the role of this term is completely understood, and omitting it makes it easier to see the relationship with leverage.

As Equation (2.4) shows, the prediction variance depends on the calibration explanatory variables via the leverage. We will plot the squared prediction error against the leverage. There are three ways to treat the calibration set in the simulations.

(1) All N replicates use one fixed calibration set of n observations, that contains one set of explanatory variables and one set of response variables.

(2) Every replicate uses the same set of explanatory variables, but generates new errors for the response variable each time.

(3) Each replicate consists of a new sample of both explanatory variables and response variables.

2.2.2 Ordinary Least Squares Prediction Uncertainty Simulation

Simulation 2.1. Ordinary Least Squares Simulation Study

In this section, we run the simulations for the ordinary least squares regression under more realistic conditions. It is sensible to use the calibration set to make inference for a prediction set drawn from the same distribution. The simulation can be generalised to the case of multivariate normal distribution. The simulation routine is planned as below.

1. The simulation of calibration sets

   The values of β_0 and $\boldsymbol{\beta}$ are fixed. The calibration explanatory variables $\dot{\mathbf{X}}_c$ are generated independently from a standard normal distribution. This simple variance structure for $\dot{\mathbf{X}}_c$ will suffice to make the points we wish to make about ordinary least squares regression. The noise $\boldsymbol{\xi}_c$ is generated independently and identically distributed as normal with mean 0 and variance σ_ξ^2. The calibration observations are calculated as $\dot{\mathbf{y}}_c = \beta_0 + \dot{\mathbf{X}}_c\boldsymbol{\beta} + \boldsymbol{\xi}_c$.

2. The simulation of prediction sets

   The predictors $\dot{\mathbf{X}}_p$, like $\dot{\mathbf{X}}_c$, are identically and independently generated from the standard normal distribution. The prediction observations can be expressed as $\dot{\mathbf{y}}_p = \beta_0 + \dot{\mathbf{X}}_p\boldsymbol{\beta} + \boldsymbol{\xi}_p$. For a prediction sample $\mathbf{x}_p$, the leverage h=$\mathbf{x}_p(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_p'$ on average equals to $\frac{k-1}{n}$, (Belsey et al. (1980) Pages 17 and 66).

3. The calibration and the prediction

   Ordinary least squares estimation gives the estimated regression coefficients $\hat{\beta}_0 = \bar{\dot{y}} - \bar{\dot{\mathbf{x}}}\hat{\boldsymbol{\beta}}$ and $\hat{\boldsymbol{\beta}} = (\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{X}_c'\mathbf{y}_c$. A prediction can be calculated as $\hat{\dot{y}}_p = \bar{\dot{y}} + \mathbf{x}_p\hat{\boldsymbol{\beta}}$.

The numerical experiment $k = 3$, $\beta_0 = 1$, $\boldsymbol{\beta} = \mathbf{1}$, $N = 100,000$, $n = 100$, $n_p = 200$, $\sigma_\xi^2 = 0.25$, and $\boldsymbol{\xi}_p = \mathbf{0}$. The average of leverages and squared prediction errors are taken by the Chi-square Binning Method described below, motivated by the fact that the leverage has a Chi-square distribution with k degrees of freedom. The number of bins is set to be 20.

Definition 2.1. Chi-square Binning Method

Denote the number of bins by b. The bins are formed to contain equal probability $\frac{1}{b}$, with reference to the χ_k^2 distribution of the leverage. The arithmetic series of the cumulative probabilities are 0, $\frac{1}{b}$, $\frac{2}{b}$, $\cdots$, $\frac{b-1}{b}$, 1. So, denote the chi-square variable values with respect to these probabilities as

$$\boldsymbol{\chi} = (\ 0 \quad \chi_1 \quad \chi_2 \quad \chi_3 \quad \cdots \quad \chi_{b-1} \quad \infty\ )'.$$

The leverage grid defining the bins can be calculated as $\frac{\bar{h}}{k}\boldsymbol{\chi}$, where $\bar{h}$ is the average leverage for all the prediction samples. The mean of the χ^2_k distribution equals to k, so the ratio $\frac{\bar{h}}{k}$ is an adjustment factor scaling the leverage grid. After putting the prediction samples into these bins by leverage, average leverage and average squared prediction error are taken in each bin. The chi-square binning method will result in roughly equal numbers of observations in each bin, allowing the average results to sketch a true relationship between squared prediction error and leverage.

In Figure 2.1, the red points presents average squared prediction error against average leverage, the distribution of the leverage can be seen from how the red points spread out. In Figure 2.1(a), the line of the red points is curved, so we are going to explain the curvature.

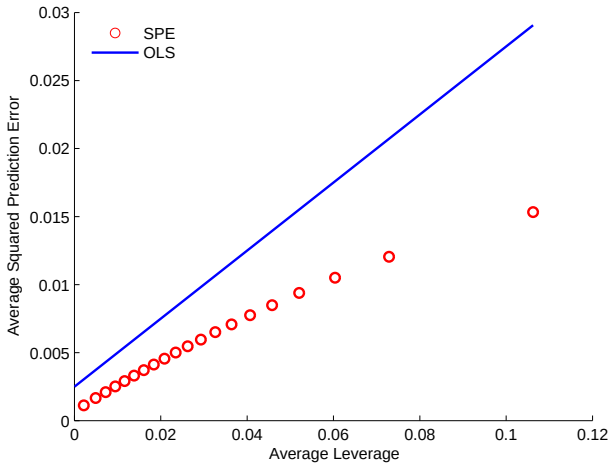
For a prediction sample, the squared prediction error can be calculated as below.

$$\begin{aligned}
\hat{\xi}_p^2 &= (\dot{y}_p - \hat{\dot{y}}_p)^2 = \{\xi_p + (\beta_0 - \hat{\beta}_0) + \mathbf{x}_p(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})\}^2 \\
&= \xi_p^2 + (\beta_0 - \hat{\beta}_0)^2 + \underbrace{\mathbf{x}_p(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})'\mathbf{x}_p'}_{\text{or } (\boldsymbol{\beta}-\hat{\boldsymbol{\beta}})'\mathbf{x}_p'\mathbf{x}_p(\boldsymbol{\beta}-\hat{\boldsymbol{\beta}})} \\
&+ 2(\beta_0 - \hat{\beta}_0)\mathbf{x}_p(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}) + 2\xi_p(\beta_0 - \hat{\beta}_0) + 2\xi_p\mathbf{x}_p(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}). \qquad (2.5)
\end{aligned}$$

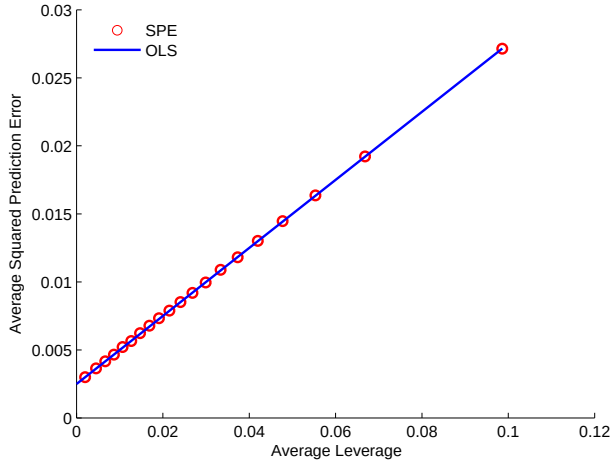
According to the ordinary least squares regression theory,

$$\begin{aligned}
\mathrm{E}\,(\hat{\xi}_p) &= 0, \\
\mathrm{E}\,(\hat{\beta}_0) &= \beta_0, \\
\mathrm{E}\,(\hat{\boldsymbol{\beta}}) &= \boldsymbol{\beta}, \\
\mathrm{Var}\,(\hat{\beta}_0) &= \mathrm{E}\,[\{\hat{\beta}_0 - \mathrm{E}\,(\hat{\beta}_0)\}^2] = (\hat{\beta}_0 - \beta_0)^2, \\
\mathrm{Var}\,(\hat{\boldsymbol{\beta}}) &= \mathrm{E}\,[\{\hat{\boldsymbol{\beta}} - \mathrm{E}\,(\hat{\boldsymbol{\beta}})\}\{\hat{\boldsymbol{\beta}} - \mathrm{E}\,(\hat{\boldsymbol{\beta}})\}'] = \mathrm{E}\,\{(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})'\} \\
&= \sigma_\xi^2(\mathbf{X}_c'\mathbf{X}_c)^{-1}, \qquad \text{as shown in Equation (2.2).}
\end{aligned}$$

Taking expectation of Equation (2.5) results in the ordinary least squares predic-

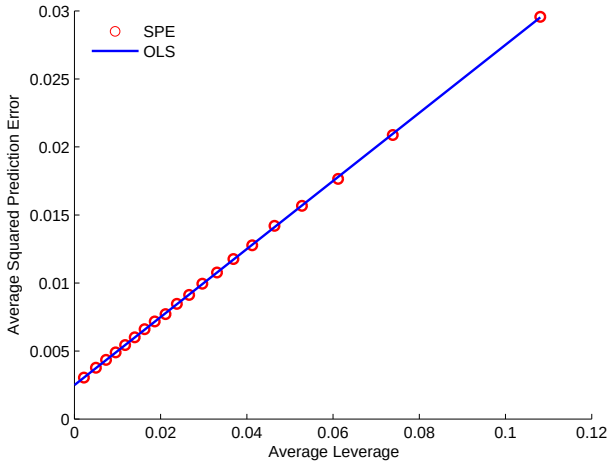


(a) Case (1) one $(\dot{\mathbf{X}}_c, \dot{\mathbf{y}}_c)$



(b) Case (2) one $\dot{\mathbf{X}}_c$, different $\dot{\mathbf{y}}_c$

Figure 2.1: OLS Average Squared Prediction Error versus Average Leverage, i.i.d. standard normally distributed predictors. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$ against average leverage $h = \mathbf{x}_p(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_p'$. OLS: the ordinary least squares prediction variance $\mathrm{Var}\,(\dot{y}_p - \hat{\dot{y}}_p) = \sigma_\xi^2(\frac{1}{n} + h)$.



(c) Case (3) different $\dot{\mathbf{X}}_c$, different $\dot{\mathbf{y}}_c$

Figure 2.1: OLS Average Squared Prediction Error versus Average Leverage, i.i.d. standard normally distributed predictors (cont.).

tion variance formula

$$\begin{aligned} \mathrm{E}\,(\hat{\xi}_p^2) &= \mathrm{E}\,\{(y_p - \hat{y}_p)^2\} \\ &= \sigma_\xi^2 + \frac{\sigma_\xi^2}{n} + \sigma_\xi^2 \underbrace{\mathbf{x}_p(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_p'}_{h}, \end{aligned}$$

where the leverage is defined as $\mathrm{E}\,\{\mathbf{x}_p(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})'\mathbf{x}_p'\}$. Hence, the ordinary least squares prediction variance is the expectation over the distribution of the estimated regression coefficients. To see the linear relationship between squared prediction error and leverge in a numerical experiment, it is required to repetitively simulate the response variable at least.

In Case (1), for fixed $\dot{\mathbf{y}}_c$, $\hat{\beta}_0$ and $\hat{\boldsymbol{\beta}}$ are fixed. The expectation of Equation (2.5) can be written as

$$E(\hat{\xi}_p^2) = \sigma_\xi^2 + (\beta_0 - \hat{\beta}_0)^2 + \mathbf{x}_p(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})'\mathbf{x}_p' + 2(\beta_0 - \hat{\beta}_0)\mathbf{x}_p(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}). \quad (2.6)$$

Except the case $k = 1$, $\mathrm{E}\,(\hat{\xi}_p^2)$ is not linear with the leverage, which explains why the red points sketch a curve in Figure 2.1(a).

Figure 2.1(b) and (c) look alike. The red points lie nicely on the blue line, verifying the linear relationship between prediction variance and leverage shown

in Equation (2.4). The difference between Figure 2.1(a) and the other two plots reminds us that the ordinary least squares prediction variance formula takes expectation over repeatedly sampled calibration sets, or at least repeatedly sampled response variables.

The range of leverages in Figure 2.1(c) is a little wider than that in Figure 2.1(b). The leverages in Figure 2.1(b) result from one realisation of $\dot{\mathbf{X}}_c$, so their range can be either wider or narrower than that of the average over $100,000$ realisations of $\dot{\mathbf{X}}_c$.

Because Figure 2.1(a) and (b) are results of the same values of explanatory variables, the ordinary least squares prediction variances calculated are the same, so the two blue lines are identical. The fact that the two lines in Figure 2.1(a) do not coincide has implications for any investigation which aims to assess the performance of an approach to quantifying uncertainty in predictions.

Simulation 2.1 has shown that the performance of the ordinary least squares regression prediction variance formula cannot be assessed from a fixed dataset. We will further show the assessment problem in Simulation 2.3. Before that, we will discuss why we do not use an obvious setup, uniformly distributed leverage, for the numerical experiment.

Simulation 2.2. Uniformly Distributed Leverage

The obvious choice, to sample predictors from the calibration explanatory variable distribution, which we will take to be multivariate normal, would lead to a poor representation of large leverages. Ideally, a uniformly distributed leverage would give a better plot. However, in this section we will use Case (1) a fixed calibration set $(\dot{\mathbf{X}}_c, \dot{\mathbf{y}}_c)$ to show that, the uniformly distributed leverage is not appropriate, as it distorts the true relationship between squared prediction error and leverage.

Except the simulation of the predictors, all other simulation procedures are carried out as the same as Simulation 2.1. To have uniformly distributed leverages we start by generating $\mathbf{z}_j$ $(1 \times k)$ from independent identically distributed standard normal distributions for the j-th prediction sample. A variable $\dot{\mathbf{u}}_j$ $(1 \times k)$ uniformly

distributed on the surface of a unit sphere can be calculated as $\dot{\mathbf{u}}_j = (\frac{z_{j1}}{Z}, \ldots, \frac{z_{jk}}{Z})$, where $Z = (\sum_{l=1}^{k} z_{jl}^2)^{\frac{1}{2}}$ (Rubinstein (1982)). The predictor $\dot{\mathbf{x}}_{p_j}$ $(1 \times k)$ for the j-th prediction sample is then calculated as $\dot{\mathbf{x}}_{p_j} = r_j \dot{\mathbf{u}}_j$, where the radius r_j is the j-th element of $\mathbf{r}$ $(1 \times n_p)$, which contains the square roots of a sequence of numbers starting from 0 and ending at $2k$ with a step size of $\frac{2k}{n_p - 1}$. Since $\frac{1}{n-1}\mathbf{X}_c'\mathbf{X}_c$ is the covariance of a sample from $N(0, \mathbf{I}_k)$, $\mathrm{E}\,(\frac{1}{n-1}\mathbf{X}_c'\mathbf{X}_c) = \mathbf{I}_k$, so $(\mathbf{X}_c'\mathbf{X}_c)^{-1} \approx \frac{1}{n}\mathbf{I}_k$. The leverage $h_j = \mathbf{x}_{p_j}(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_{p_j}' \approx \mathbf{x}_{p_j}(\frac{1}{n}\mathbf{I}_k)\mathbf{x}_{p_j}' = \frac{1}{n}\mathbf{x}_{p_j}\mathbf{x}_{p_j}' = \frac{r_j}{n}$. The observations in the prediction set can be calculated as $\dot{\mathbf{y}}_p = \beta_0 + \dot{\mathbf{X}}_p\boldsymbol{\beta}$. Hence, the leverage has approximately a uniform distribution $Unif(0, \frac{2k}{n})$, from which we construct a uniform binning method defined as below to obtain average leverage and average squared prediction error.

Definition 2.2. Uniform Binning Method

For a uniform distribution $Unif(\theta_1, \theta_2)$, the number of bins is set to be b. The bins are formed to contain equal probability $\frac{1}{b}$. The uniform variable values with respect to these probabilities as

$$\boldsymbol{\theta} = \{\ \theta_1,\ \ (1 - \tfrac{1}{b})\theta_1 + \tfrac{1}{b}\theta_2,\ \ (1 - \tfrac{2}{b})\theta_1 + \tfrac{2}{b}\theta_2,\ \ \ldots,\ \ \tfrac{1}{b}\theta_1 + (1 - \tfrac{1}{b})\theta_2,\ \ \theta_2\ \},$$

which defines the leverage grid. After putting the prediction samples into these bins by leverage, we take average leverage and average squared prediction error in each bin. The uniform binning method will give roughly equal numbers of observations in each bin. In our case, $\theta_1 = 0$, $\theta_2 = \frac{2k}{n}$.

The numerical study begins with $k = 3$, $\beta_0 = 1$, $\boldsymbol{\beta} = \mathbf{1}$, $N = 100,000$, $n_p = 21$, $\sigma_\xi^2 = 0.25$, and $\boldsymbol{\xi}_p = \mathbf{0}$. We run two experiments: one has a calibration set with a size of 100, and the other has 10000 observations in the calibration set. We set up the number of bins in the uniform binning method to be 20. An average squared prediction error against average leverage plot is used to show the relationship as shown in Equation (2.4), that describes the dependence of prediction error on $\mathbf{x}_p$ (via h). To avoid noise, we use the true value of σ_ξ^2 in the ordinary least squares prediction variance formula rather than use an estimate. As $\boldsymbol{\xi}_p = \mathbf{0}$, the linear

relationship that should be reproduced by the simulation can be written as

$$\text{Var}\,(\dot{y}_p - \hat{\dot{y}}_p) = \sigma_\xi^2(\frac{1}{n} + h). \tag{2.7}$$

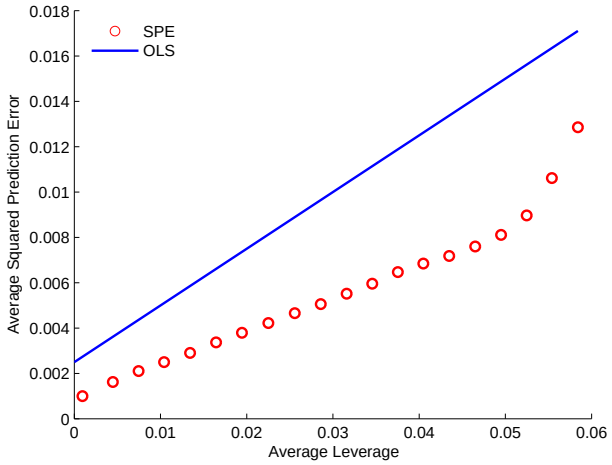
In Figure 2.2, the blue line shows the linear relationship described in Equation (2.7), and the red points display the actual relationship between the squared prediction error and the leverage observed in the simulation. In Figure 2.2(a), the blue line has an intercept of $\frac{\sigma_\xi^2}{n} = 0.0025$, and the slope of $\sigma_\xi^2 = 0.25$. In Figure 2.2(b), the intercept of the blue line is equal to $\frac{\sigma_\xi^2}{n} = 0.000025$, and its slope is still equal to $\sigma_\xi^2 = 0.25$.

As the squared prediction error $\hat{\xi}_{p_j}^2$ is proportional to r_j, and the leverage $h_j \approx \frac{r_j}{n}$, the red points are supposed to form a straight line. But in Figure 2.2(a) the last four red points are bent up. The curvature appears because $\frac{1}{n}\mathbf{I}_k$ does not give a good approximation of $(\mathbf{X}_c'\mathbf{X}_c)^{-1}$ when $n = 100$, which causes the curvature at the tail. In other words, if the sample size is large, the approximation would work well. The leverage has a better uniform distribution. It is why in Figure 2.2(a) the red points are perfectly linear when $n = 10000$.

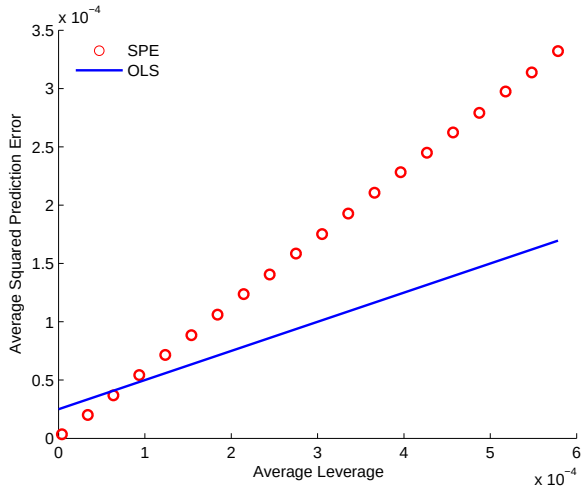
As for a prediction sample $\mathbf{x}_{p_j}$, the uniformly distributed leverage design enforces both squared prediction error and leverage to be proportional to r_j. In fact, for a fixed calibration set, the true relationship between squared prediction error and leverage may not be linear, which has been discussed in Simulation 2.1 where Figure 2.1(a) is a typical example. Hence, the uniformly distributed leverage design distorts the true relationship between squared prediction error and leverage for a fixed calibration set. Moreover, for principal components regression and partial least squares regression, the distribution of the predictors will affect the relationship between the bias and the leverage, so the simulations with uniformly distributed leverage would fix the pattern of this relationship in an unnatural way. Hence, the predictors with a multivariate normal distribution would be more appropriate.

Simulation 2.3. Three Different Simulations of $\mathbf{y}_c$ for a Fixed $\mathbf{X}_c$

We run a numerical experiment Case (1) three times, each of which contains



(a) $n = 100$



(b) $n = 10000$

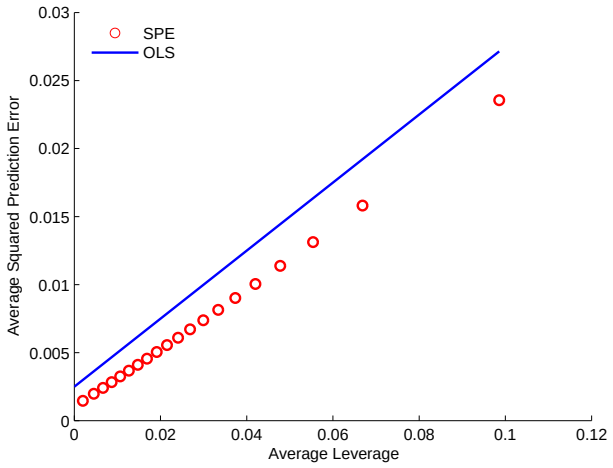
Figure 2.2: OLS Average Squared Prediction Error versus Average Leverage, uniformly distributed leverage, Case (1) a fixed calibration set $(\dot{\mathbf{X}}_c, \dot{\mathbf{y}}_c)$. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$ against average leverage $h = \mathbf{x}_p(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_p'$. OLS: the ordinary least squares prediction variance, $\text{Var}\,(\dot{y}_p - \hat{\dot{y}}_p) = \sigma_\xi^2(\frac{1}{n} + h)$ (See Equation (2.7)).

$100,000$ replicates. Each experiment has a set of explanatory variables, but different replicates of the response variable are simulated for the fixed calibration set. The parameters are set as the same as Simulation 2.1. There are three plots displayed in Figure 2.3, they compare the relationship of observed squared prediction error against leverage and theoretical prediction variance against leverage. The red points present average squared prediction error against average leverage for the particular set of the response variable. The blue line presents the ordinary least squares prediction variance given by Equation (2.7).

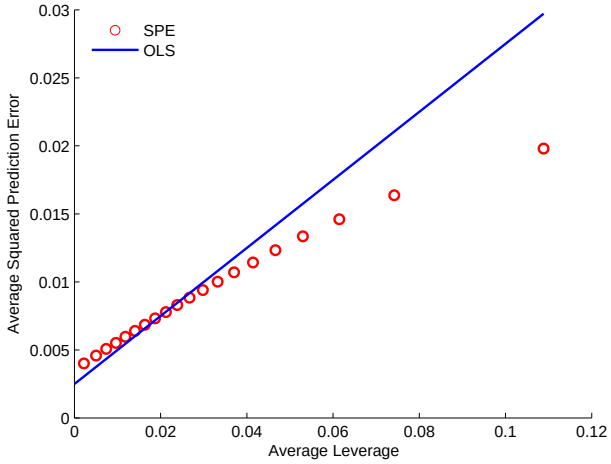
In Figure 2.3(a), the slope of the red line is similar to the blue line; in Figure 2.3(b) the slope of the red line is smaller than that of the blue line; in Figure 2.3(c) the slope of the red line is bigger than that of the blue line. The red lines can be steeper, flatter, or similar to the blue lines. This is because red points in the three graphs are drawn for three different simulations of the response variable in the calibration sets. And, the curvature formed by the red points is decided by the distribution of explanatory variables in the calibration set, which has been explained in Simulation 2.1.

The blue line in Figure 2.3 is an expected value over repeated sampling of $\mathbf{y}_c$, and does not describe the behavior of the errors for fixed $\mathbf{y}_c$. The performance of the ordinary least squares regression prediction variance formula cannot be assessed in the obvious way referring to a fixed dataset, because the prediction variance formula takes expectation over the distribution of the estimated regression coefficients, and the squared prediction error calculated for a single set of data always relies on the estimated regression coefficients from this dataset. It is important to be aware of this behavior of the prediction variance formula in order to study prediction uncertainty of principal components regression and partial least squares regression.

Additional problems that will arise with real data sets are the availability of prediction samples, and the presence of noise in the predictions. Each red point in the plot of Figure 2.3 is an average over $100,000$ squared prediction errors against their leverages, and the regression variance in the prediction set σ_ξ^2 is set to be

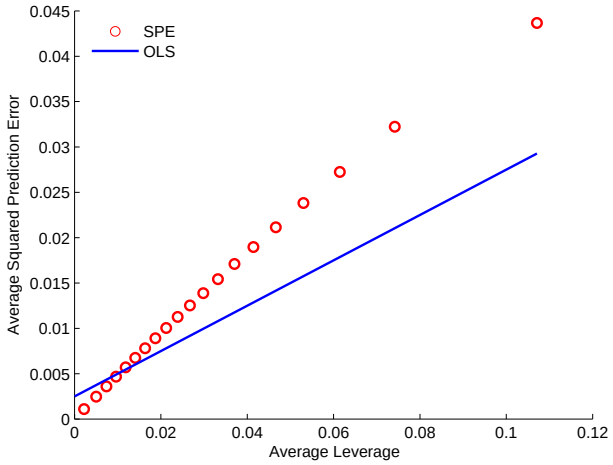


(a) First simulation of $\dot{\mathbf{y}}_c$ for a fixed $\dot{\mathbf{X}}_c$



(b) Second simulation of $\dot{\mathbf{y}}_c$ for a fixed $\dot{\mathbf{X}}_c$

Figure 2.3: OLS Average Squared Prediction Error versus Average Leverage, uniformly distributed leverage, three simulations of $\mathbf{y}_c$ for a fixed $\mathbf{X}_c$. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$ against average leverage $h = \mathbf{x}_p(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_p'$. OLS: the ordinary least squares prediction variance, $\mathrm{Var}\,(\dot{y}_p - \hat{\dot{y}}_p) = \sigma_\xi^2(\frac{1}{n} + h)$.



(c) Third simulation of $\dot{\mathbf{y}}_c$ for a fixed $\dot{\mathbf{X}}_c$

Figure 2.3: OLS Average Squared Prediction Error versus Average Leverage, uniformly distributed leverage, three simulations of $\mathbf{y}_c$ for a fixed $\mathbf{X}_c$ (cont.).

zero. For a real dataset, it is difficult to collect the data with such a large number of predictions, and the noise term always exists. Hence, the red points are unlikely to form such a clear straight line.

With real data sets, we need to find some way to round the assessment difficulty. We will try to model the linear relationship shown in the ordinary least squares prediction variance formula empirically, via the use of the tuning set (Section 2.2.3) or cross-validation (Section 2.2.4). And then, we will investigate whether random data splitting (Section 2.2.5) can be used to round the assessment difficulty or not.

2.2.3 The Use of a Tuning Set

In Section 1.1 we have shown that a tuning set $(\dot{\mathbf{X}}_t, \dot{\mathbf{y}}_t)$ sampled from the same distribution as the calibration set can be useful in the quantification of prediction uncertainty in the ordinary least squares regression, where the number of observations in the tuning set is denoted as n_t. After we obtain the estimated regression coefficients from the calibration set, the tuning set can be used in two different ways. One is to calculate the root mean squared error of prediction (Equation (1.8)) as an empirical estimate of prediction error. The other way is to use the

estimate of regression error variance (Equation (1.9)) from the tuning set in the ordinary least squares prediction variance formula. The tuning set can also be used as the same way in the study of principal components regression and partial least squares regression. In this section, we would like to study whether a tuning set can be used to provide a sensible empirical estimate of the ordinary least squares prediction variance formula, so that it can provide an approximation to the prediction variance for a real dataset. We hope this would shed some light on using the tuning set in principal components regression and partial least squares regression.

For a real dataset, the calibration set is fixed. Although in this case the relationship between squared prediction error and leverage may not be linear as shown in Equation (2.6) Simulation 2.1, the ordinary least squares prediction variance formula can be used as an approximation to prediction variance. Since the ordinary least squares prediction variance formula gives a linear relationship between squared prediction error and leverage, would it be possible to quantify the approximate predication variance through estimating the slope and the intercept of the formula? If possible, how many samples are needed in the tuning set in order to give reasonable estimates of the slope and the intercept? Is there any relationship between the sample sizes n and n_t? To answer these questions, we consider the relationship between the squared prediction error $\hat{\boldsymbol{\xi}}_t^2$ and the leverage $\mathbf{h}_t$ calculated in the tuning set.

For a calibration set $(\dot{\mathbf{X}}_c, \dot{\mathbf{y}}_c)$ and a tuning set, $\dot{\mathbf{X}}_c \sim N(\mathbf{0}, \boldsymbol{\Sigma})$ and $\dot{\mathbf{X}}_t \sim N(\mathbf{0}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma}$ is the covariance matrix. Assume $\boldsymbol{\Sigma} = \mathbf{C}\mathbf{C}'$, thus $(\mathbf{X}_c'\mathbf{X}_c)^{-1} \approx \frac{1}{n}(\mathbf{C}\mathbf{C}')^{-1}$. For a sample in the tuning set, the transformation of the centred predictor $C^{-1}\mathbf{x}_t' \sim N(\mathbf{0}, \boldsymbol{\Sigma})$, which gives $\mathbf{x}_t(CC')^{-1}\mathbf{x}_t'$ has a Chi-square distribution with k degrees of freedom.

$$\mathbf{x}_t(C')^{-1}C^{-1}\mathbf{x}_t' = \mathbf{x}_t(CC')^{-1}\mathbf{x}_t' \sim \chi_k^2.$$

The leverage $h_t = \mathbf{x}_t(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_t' \approx \mathbf{x}_t\frac{1}{n}(\mathbf{C}\mathbf{C}')^{-1}\mathbf{x}_t'$, which has approximately a Chi-square distribution $\frac{1}{n}\chi_k^2$. $\mathrm{E}(h_t) = \frac{k}{n}$ and $\mathrm{Var}(h_t) = \frac{2k}{n}$. $\mathrm{E}\{\sum_{i=1}^{n_t}(h_{t_i} - \bar{h}_t)^2\} \approx \frac{2kn_t}{n}$, where $\bar{h}_t = \frac{1}{n_t}\sum_{i=1}^{n_t} h_{t_i}$.

The ordinary least squares prediction variance formula gives a linear relationship between leverage and prediction variance. We could use this relationship to build a linear model to regress squared prediction error on leverage for the tuning set, thus giving empirical estimates how prediction variance associates with leverage. Assume the simple linear model as

$$\hat{\boldsymbol{\xi}}_t^2 = \omega_0 + \omega_1 \mathbf{h}_t + \boldsymbol{\nu}, \tag{2.8}$$

where $\hat{\boldsymbol{\xi}}_t^2$ denotes the squared prediction errors, $\mathbf{h}_t$ denotes the leverages in the tuning set, and $\boldsymbol{\nu}$ is the noise term. The regression coefficient ω_0 is the intercept, and ω_1 is the slope. According to the ordinary least squares regression theory, $\sigma_\nu^2 = \mathrm{Var}\,(\xi_t^2) = 2\sigma_\xi^4$. $\hat{\omega}_1 \approx \sigma_\xi^2$, so the variance of the estimated slope

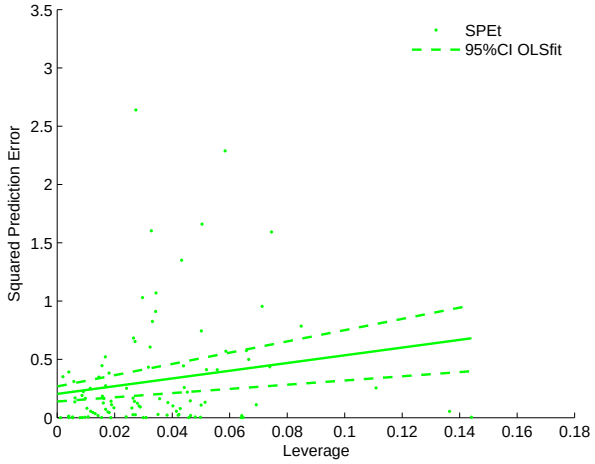
$$\mathrm{Var}\,(\hat{\omega}_1) = \frac{\sigma_\nu^2}{\sum_{i=1}^{n_t}(h_{t_i} - \bar{h}_t)^2} \approx \frac{2\sigma_\xi^4}{2kn_t/n} = \frac{\sigma_\xi^4 n}{kn_t}.$$

The coefficient of variation of $\hat{\omega}_1$

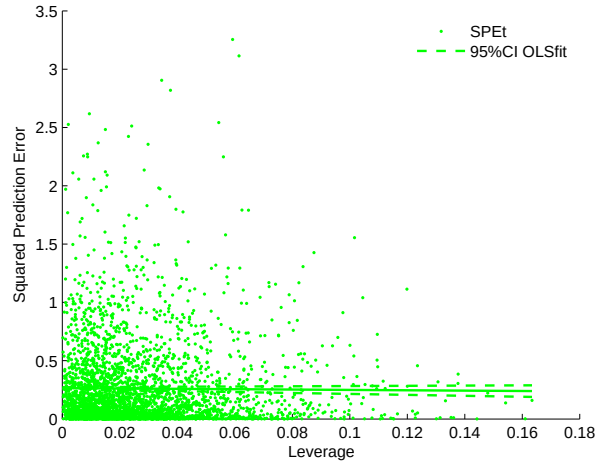
$$c.v.(\hat{\omega}_1) = \frac{\sqrt{\mathrm{Var}\,(\hat{\omega}_1)}}{\hat{\omega}_1} = \sqrt{\frac{\sigma_\xi^4 n}{kn_t}}/\sigma_\xi^2 = \sqrt{\frac{n}{kn_t}}. \tag{2.9}$$

Intuitively, for fixed n and n_t, when there are more predictors, the leverage increases on average. If we study squared prediction error at a certain level, when larger leverages account more for the deterministic part of the linear model (See Equation (2.8)), the estimated regression coefficient $\hat{\omega}_1$ is more stable. Usually $\hat{\omega}_1$ is good enough as an estimate of the slope when $c.v.(\hat{\omega}_1)$ is less than or equal to 0.1000 in magnitude. We often use ordinary least squares regression when the number of explanatory variables less than or equal to 7. When $k = 7$, $n = 10$ and $n_t = 100$, Equation (2.9) gives $c.v.(\hat{\omega}_1) = 0.1195$, which suggests the size of the tuning set should be at least 10 times of the size of the calibration set. In the real situation, it is difficult to obtain such a large size tuning set. When the calibration set and tuning set have equal size, $c.v.(\hat{\omega}_1) \leq 0.1000$ requires $k \geq 100$. This is unrealistic because when there are more than 10 explanatory variables we usually do not employ ordinary least squares regression.

In Simulation 2.1 Case (1) when $k = 3$, suppose there is a tuning set that has equal size with the calibration set $n_t = n = 100$, $c.v.(\hat{\omega}_1) = \sqrt{\frac{1}{3}} \approx 0.5774$. When $n_t \geq 3333$, $c.v.(\hat{\omega}_1)$ can attain 0.1000.



(a) $k = 3$, $n = 100$ and $n_t = 100$



(b) $k = 3$, $n = 100$ and $n_t = 3333$

Figure 2.4: OLS Squared Prediction Error versus Leverage in a Tuning Set, i.i.d. standard normally distributed predictors, one $(\dot{\mathbf{X}}_c, \dot{\mathbf{y}}_c)$. SPEt: squared prediction error in the tuning set $(\dot{y}_t - \hat{\dot{y}}_t)^2$ against leverage $h_t = \mathbf{x}_t(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_t'$. 95%CI OLSfit: 95% confidence interval of the ordinary least squares fit of all squared prediction errors against leverages in the tuning set.

Figure 2.4 displays the result of an experiment of Simulation 2.1 Case (1), $k = 3$, $n = 100$, where the green point denotes squared prediction error against leverage in a tuning set. The two green dash lines give the 95% confidence interval of the ordinary least squares fit of all squared prediction error against leverage in the tuning set. The solid green line is drawn by $\hat{\boldsymbol{\xi}}_t^2 = \hat{\omega}_0 + \hat{\omega}_1 \mathbf{h}_t$. Figure 2.4(a) plots the case when $n_t = 100$ and (b) presents the case when $n_t = 3333$. The 95% confidence interval band in (a) is much wider than that in (b) given the two plots are drawn on the same scales. This suggests that when $n_t = 100$ the estimated slope is more variable than that when $n_t = 3333$, which is consistent with the coefficient variations where (a) $c.v.(\hat{\omega}_1) \approx 0.5774$ and (b) $c.v.(\hat{\omega}_1) \approx 0.1000$.

In theory, the ordinary least squares prediction variance formula can be estimated empirically by conducting a simple linear regression of squared prediction error against leverage in a tuning set. However, the realisation of this method requires a minimum size of the tuning set according to the number of explanatory variables and the sample size of the calibration set. In practice, the number of explanatory variables, the sample sizes of the calibration and the tuning sets cannot coordinate easily to generate reliable regression coefficient estimates for the ordinary least squares prediction variance formula. Hence, it would be difficult to calculate the approximate prediction variance for a real dataset in this way. It is unnecessary to study the tuning set like this in principle components regression and partial least squares regression.

2.2.4 Cross-validation

Before studying principal components regression and partial least squares regression, we study cross-validation in ordinary least squares regression. The cross-validation can be used in the same way as the tuning set. From the cross-validation, we are able to calculate an empirical estimate of prediction error and an estimated regression error variance, which also applies in principal components regression and partial least squares regression. We present how to carry out the cross-validation as below.

Definition 2.3. Cross-validation

Leave-one-out cross-validation builds reduced data sets by deleting one observation each time. $(\dot{\mathbf{X}}_{c_{-j}}, \dot{\mathbf{y}}_{c_{-j}})$ is the new dataset constructed by leaving out the j-th observation $(\dot{\mathbf{x}}_{c_j}, \dot{y}_{c_j})$, that is,

$$\begin{aligned} \dot{\mathbf{X}}_{c_{-j}} &= (\dot{\mathbf{x}}_{c_1}, \ldots, \dot{\mathbf{x}}_{c_{j-1}}, \dot{\mathbf{x}}_{c_{j+1}}, \ldots, \dot{\mathbf{x}}_{c_n}) \\ \dot{\mathbf{y}}_{c_{-j}} &= (\dot{y}_{c_1}, \ldots, \dot{y}_{c_{j-1}}, \dot{y}_{c_{j+1}}, \ldots, \dot{y}_{c_n})' \qquad j = 1, \ldots, n \end{aligned}$$

After the new dataset is generated, the leave-out observation is used as the prediction.

When it comes to estimating empirically the approximate prediction variance presented by the ordinary least squares prediction variance formula for a real dataset, cross-validation can be taken as the case when using a tuning set with $n_t = n$, so the coefficient variation $c.v.(\hat{\omega}_1) = \sqrt{\frac{1}{k}}$ (See Equation (2.9)). If we set $c.v.(\hat{\omega}_1) \leq 0.1000$, the number of explanatory variables needs to be at least 100. A large number of explanatory variables ensures $\hat{\omega}_1$ to be estimated properly as $c.v.(\hat{\omega}_1)$ is very small in this case. However, it does not make sense to apply ordinary least squares regression for such a large number of explanatory variables. Likewise, we will not use cross-validation in this way for principal components regression and partial least squares regression.

2.2.5 Random Data Splitting

In this section, we will investigate whether random data splitting can be a useful tool or not, to round the problem that the performance of the ordinary least squares prediction variance formula cannot be assessed by any single data sets directly. If we want to use random data splitting in the study of principal components regression and partial least squares regression, it should work in ordinary least squares regression.

Simulation 2.4. Random Data Splitting for One Set of Simulated Data

There is only one set of data $(\dot{\mathbf{X}}_c, \dot{\mathbf{y}}_c)$ simulated, which contains $n + n_t + n_p$ observations. We run N replicates, in each of which the simulated the data is

randomly permuted and split into a calibration set with n observations, a tuning set with n_t observations, and a prediction set with n_p observations. The three data sets are exchangeable. Estimated regression coefficients are calculated from the calibration set, and are used to make predictions for the tuning set and the prediction set. Squared prediction errors and leverages in both of the tuning set and the prediction set are saved to be compared.

In order to investigate how the noise of the dataset make an influence on the result, we fix the noise term in the prediction set to be $\boldsymbol{\xi}_p = 0.25$. The regression model can be written as

$$\begin{aligned} \dot{\mathbf{y}}_c &= \beta_0 + \dot{\mathbf{X}}_c\boldsymbol{\beta} + \boldsymbol{\xi}_c, \\ \dot{\mathbf{y}}_t &= \beta_0 + \dot{\mathbf{X}}_t\boldsymbol{\beta} + \boldsymbol{\xi}_t, \\ \dot{\mathbf{y}}_p &= \beta_0 + \dot{\mathbf{X}}_p\boldsymbol{\beta} + \boldsymbol{\xi}_p. \end{aligned}$$

The parameters of the numerical experiment are configured as following. $N = 100000$, $k = 1$, $\dot{\mathbf{X}}_c \sim N(0,1)$, $\beta_0 = \beta_1 = 1$, $\sigma_\xi^2 = 0.25$, $n = 200$, $n_t = 200$, and $n_p = 200$.

In Figure 2.5, the green point (SPEt) stands for average squared prediction error against average leverage calculated from the tuning set. The green line (SPEt fit) is the ordinary least squares fit of all squared prediction error against leverage in the tuning set. The pink point (SPE) presents average squared prediction error against average leverage. The pink dash line (SPE fit) is the ordinary least squares fit of all squared prediction error against leverage. The light blue line (OLS) is given by the ordinary least squares regression variance where $\sigma_\xi^2 = 0.25$ for simplicity.

The pink points form a straight line (SPE fit). The light blue line (OLS) and the pink dash line (SPE fit) overlap, because the error term is fixed as 0.25. The green points (SPEt) are so noisy that the green line (SPEt OLSfit) is quite different from the blue and pink lines.

If the error term in the prediction set is not fixed, the tuning set and the prediction set would have exactly the same result because the tuning set and the prediction set are exchangeable. The noisy green points Figure 2.5 suggests it is

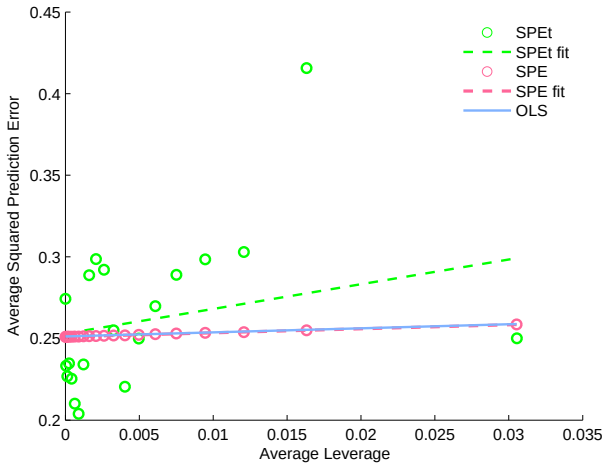


Figure 2.5: OLS Average Squared Prediction Error versus Average Leverage for One Set of Simulated Data, random data splitting, $\boldsymbol{\xi}_p = 0.25$. SPEt: average squared prediction error in the fit of the tuning set $(\dot{y}_t - \hat{\dot{y}}_t)^2$ against average leverage $h_t = \mathbf{x}_t(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_t'$. SPEt fit: the ordinary least square fit of all squared prediction errors in the fit of the tuning set. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$ against average leverage $h = \mathbf{x}_p(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_p'$. SPE fit: the ordinary least square fit of all squared prediction errors. OLS: the ordinary least squares prediction variance $\text{Var}\,(\dot{y}_p - \hat{\dot{y}}_p) = \sigma_\xi^2(\frac{1}{n} + h + 1)$.

unable to round the problem for a real dataset through the random data splitting, because the noise in the nature of a single dataset has been systematically amplified by the random data splitting. But the tuning set can be used in the estimation of regression variance serving as an adjustment for this particular set of data.

Simulation 2.5. Random Data Splitting Simulation Study for 400, 000 Sets of Simulated Data

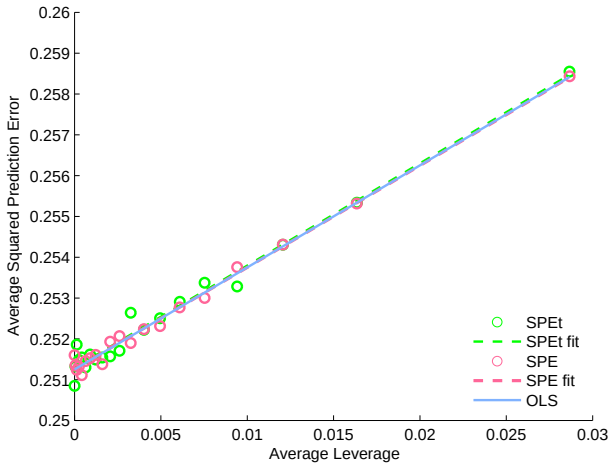


Figure 2.6: OLS Average Squared Prediction Error versus Average Leverage for 400, 000 Sets of Simulated Data, random data splitting. SPEt: average squared prediction error in the fit of the tuning set $(\dot{y}_t - \hat{\dot{y}}_t)^2$ against average leverage $h_t = \mathbf{x}_t(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_t'$. SPEt fit: the ordinary least square fit of all squared prediction errors in the fit of the tuning set. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$ against average leverage $h = \mathbf{x}_p(\mathbf{X}_c'\mathbf{X}_c)^{-1}\mathbf{x}_p'$. SPE fit: the ordinary least square fit of all squared prediction errors. OLS: the ordinary least squares prediction variance $\text{Var}\,(\dot{y}_p - \hat{\dot{y}}_p) = \sigma_\xi^2(\frac{1}{n} + h + 1)$.

In Simulation 2.2, it has been verified that the ordinary least squares prediction variance presents the average behavior. To illustrate it using random data splitting, we run 400, 000 replicates, each of which has a set of simulated data, and then it is randomly split into the calibration set, the tuning set and the prediction set. Keeping all other simulation parameters as the same as Simulation 2.4, Figure 2.6 is plotted to show the result.

The pink point (SPE) gives average squared prediction error against average leverage, and the pink dash line (SPE fit) is the ordinary least squares fit of all squared prediction error against leverage. The green point (SPEt) presents average squared prediction error against average leverage calculated from the tuning set, and the green line (SPEt fit) is the ordinary least squares fit of all squared prediction error against leverage in the tuning set. The light blue line (OLS) is drawn by the ordinary least squares regression variance.

The pink line, the green and the blue line overlap. The pink points and green points are noisy, and give different pattern, but they are fitted to the overlapped lines. The result is consistent with what we have seen in Figure 2.1(c) that the ordinary least squares prediction variance is the result of taking expectation over lots of different data sets.

2.3 Some Comments on Leverage

In the case of ordinary least squares, leverage plays a key role. The formula for predictive variance is linear in leverage, and plotting average squared errors against leverage is a natural way of summarising the predictive performance of a regression equation on a validation set. Once we begin to study biased regression methods such as PCR or PLS the relevance of leverage becomes less obvious. When the dimension of the vector of predictors is high we often cannot compute the leverage in the full x-space. We can compute it in a reduced space spanned by the constructed predictors, but this measure will fail to capture the effect of extrapolations in directions orthogonal to the reduced space. We can, and will, compute other measures of the distance of x from the centre of the calibration data. These are leverage-like, in the sense that they are quadratic forms in x, because they all arise from second order approximations to prediction mean squared error. These leverages will be used extensively, both as part of approximate formulas for prediction mean squared error and as convenient ways to summarise the average performance of various methods.

2.4 Summary

For ordinary least squares regression, simple or multiple, it is a standard result that prediction variance is linearly related to leverage. Investigating this known result by simulations has provided some guidance for designing later simulations for PCR and PLS, and has shown that one obvious empirical approach for estimating predictive variance in the case of these more complex methods will not work.

The first and more obvious point is that any simulations deigned to assess the performance of a prediction variance formula will need to involve repetitions of the calibration set, or at least of the response variable in the calibration set. One implication of this is that one can learn very little from applying proposed formulas to any single real data set. With a fixed calibration set, the estimated slope vector is fixed, and what was variance in the prediction formula becomes bias. Repeated splitting of a fixed data set does not overcome this problem because the estimated slope vectors from the various calibration sets are still biased towards that from the full data set.

The second conclusion relates to the obvious idea of trying to use a second set of data, called here a tuning set, to estimate empirically the relationship between prediction mean squared error and some measure of the distance of the x-vector from the calibration set mean. This idea fails with OLS regression because the relationship is too weak compared with the noise in the predictions, and so is not worth exploring for PCR or PLS. There is also an implication here for the possibility of assessing prediction mean squared error formulas on any real data set: the prediction set would need to be impossibly large to give any useful information.

Chapter 3

Principal Components Regression Prediction Uncertainty

Principal components regression theory is presented in Section 3.1. Section 3.2 introduces empirical estimates and ordinary least squares type prediction mean squared error, two common approaches to evaluate the prediction performance. In Section 3.3, we use simulation studies to investigate the quantification of prediction uncertainty in principal components regression step by step.

- Section 3.3.1 uses independent normally distributed explanatory variables in the principal components regression simulation to reveal there is a discrepancy in the ordinary least squares type prediction mean squared error and the true relationship between squared prediction error and leverage.
- Section 3.3.2 demonstrates the discrepancy in the ordinary least squares type prediction mean squared error and the true relationship between squared prediction error and leverage is caused by the unselected explanatory variables.
- Section 3.3.3 tries to build a model associating prediction mean squared error with sample size because the sample size appears in the ordinary least squares type prediction mean squared error formula, and the leverage is also a function of the sample size.

- Section 3.3.4 provides anther approach to find how the leverage is associated with the bias. We look for a straightforward connection between the bias and the leverage, hoping the leverage could be helpful to express the variance brought by the bias.

3.1 Principal Components Regression Theory

3.1.1 Principal Components (PCs)

As shown in Jolliffe (2002), suppose $\mathbf{x}$ is a row vector of k random variables with population covariance matrix $\mathbf{\Sigma}$. The first principal component is the linear function $\mathbf{x}\boldsymbol{\theta}_1 = \sum_{j=1}^{k} \theta_{1j} x_j$ having maximum variance, where $\boldsymbol{\theta}_1$ $(k \times 1)$ is a column vector. Then we look for the linear function $\mathbf{x}\boldsymbol{\theta}_2$ uncorrelated with $\mathbf{x}\boldsymbol{\theta}_1$ and having maximum variance, and so on. A linear function $\mathbf{x}\boldsymbol{\theta}_a$ is found that has maximum variance subject to being uncorrelated with $\mathbf{x}\boldsymbol{\theta}_1, \mathbf{x}\boldsymbol{\theta}_2, \ldots, \mathbf{x}\boldsymbol{\theta}_{a-1}$. The i-th derived variable, $\mathbf{x}\boldsymbol{\theta}_i$ is the i-th PC. In general, most of the variation in $\mathbf{x}$ will be accounted for by $a < k$ PCs, and when k is large we often have $a \ll k$.

The variance of $\mathbf{x}\boldsymbol{\theta}_1$, $\mathrm{Var}\,(\mathbf{x}\boldsymbol{\theta}_1) = \boldsymbol{\theta}_1' \mathrm{Var}\,(\mathbf{x}) \boldsymbol{\theta}_1 = \boldsymbol{\theta}_1' \mathbf{\Sigma} \boldsymbol{\theta}_1$. The maximum will not be achieved for finite $\boldsymbol{\theta}_1$ so a normalisation constraint must be imposed. Here we choose $\boldsymbol{\theta}_1' \boldsymbol{\theta}_1 = 1$. To maximise $\boldsymbol{\theta}_1' \mathbf{\Sigma} \boldsymbol{\theta}_1$ subject to $\boldsymbol{\theta}_1' \boldsymbol{\theta}_1 = 1$ is equivalent to maximising the Lagrange function

$$\boldsymbol{\theta}_1' \mathbf{\Sigma} \boldsymbol{\theta}_1 - \lambda_1 (\boldsymbol{\theta}_1' \boldsymbol{\theta}_1 - 1),$$

where λ_1 is a Lagrange multiplier. To take the derivative with respect to $\boldsymbol{\theta}_1'$ gives

$$\mathbf{\Sigma} \boldsymbol{\theta}_1 - \lambda \boldsymbol{\theta}_1 = \mathbf{0}, \qquad \text{or} \qquad (\mathbf{\Sigma} - \lambda_1 \mathbf{I}_k) \boldsymbol{\theta}_1 = \mathbf{0}.$$

Thus, λ_1 is an eigenvalue of $\mathbf{\Sigma}$ and $\boldsymbol{\theta}_1$ is the corresponding eigenvector. Note that $\boldsymbol{\theta}_1' \mathbf{\Sigma} \boldsymbol{\theta}_1 = \boldsymbol{\theta}_1' \lambda_1 \boldsymbol{\theta}_1 = \lambda_1 \boldsymbol{\theta}_1' \boldsymbol{\theta}_1 = \lambda_1$. Hence $\boldsymbol{\theta}_1$ is chosen as the eigenvector corresponding to λ_1, the largest eigenvalue of $\mathbf{\Sigma}$. The second PC, $\mathbf{x}\boldsymbol{\theta}_2$, maximises $\boldsymbol{\theta}_2' \mathbf{\Sigma} \boldsymbol{\theta}_2$ subject to being uncorrelated with $\mathbf{x}\boldsymbol{\theta}_1$, or equivalently subject to $\mathrm{Cov}\,(\mathbf{x}\boldsymbol{\theta}_1, \mathbf{x}\boldsymbol{\theta}_2) = 0$. Because $\mathrm{Cov}\,(\mathbf{x}\boldsymbol{\theta}_1, \mathbf{x}\boldsymbol{\theta}_2) = \boldsymbol{\theta}_1' \mathbf{\Sigma} \boldsymbol{\theta}_2 = \boldsymbol{\theta}_1' \lambda_1 \boldsymbol{\theta}_2 = \lambda_1 \boldsymbol{\theta}_1' \boldsymbol{\theta}_2$, either

of the equations

$$\begin{aligned} \boldsymbol{\theta}_1'\boldsymbol{\Sigma}\boldsymbol{\theta}_2 &= 0, \quad \text{i.e. uncorrelated scores or} \\ \boldsymbol{\theta}_1'\boldsymbol{\theta}_2 &= 0, \quad \text{i.e. orthogonal loadings} \end{aligned}$$

can be used to specify there is no correlation between $\mathbf{x}\boldsymbol{\theta}_1$ and $\mathbf{x}\boldsymbol{\theta}_2$. Taking one as a constraint the other will follow as a property of the solution. If we arbitrarily choose orthogonal loadings as the constraint, the Lagrange function can be written as

$$\boldsymbol{\theta}_2'\boldsymbol{\Sigma}\boldsymbol{\theta}_2 - \lambda_2(\boldsymbol{\theta}_2'\boldsymbol{\theta}_2 - 1) - \phi\boldsymbol{\theta}_1'\boldsymbol{\theta}_2,$$

where λ_2 and ϕ are Lagrange multipliers. To differentiate with respect to $\boldsymbol{\theta}_2'$, the equation above gives

$$\boldsymbol{\Sigma}\boldsymbol{\theta}_2 - \lambda_2\boldsymbol{\theta}_2 - \phi\boldsymbol{\theta}_1 = \mathbf{0}.$$

To multiply this equation on both sides by $\boldsymbol{\theta}_1'$ gives

$$\boldsymbol{\theta}_1'\boldsymbol{\Sigma}\boldsymbol{\theta}_2 - \lambda_2\boldsymbol{\theta}_1'\boldsymbol{\theta}_2 - \phi\boldsymbol{\theta}_1'\boldsymbol{\theta}_1 = 0,$$

which reduces to $\phi = 0$, because the first two terms are zero and $\boldsymbol{\theta}_1'\boldsymbol{\theta}_1 = 1$. Therefore, we have $\boldsymbol{\Sigma}\boldsymbol{\theta}_2 - \lambda_2\boldsymbol{\theta}_2 = \mathbf{0}$ or equivalently, $(\Sigma - \lambda_2 I_k)\boldsymbol{\theta}_2 = \mathbf{0}$. λ_2 is an eigenvalue of Σ, and $\boldsymbol{\theta}_2$ is the corresponding eigenvector. Again, $\lambda_2 = \boldsymbol{\theta}_2'\boldsymbol{\Sigma}\boldsymbol{\theta}_2$, so λ_2 is to be as large as possible. Assuming that Σ does not have equal eigenvalues, λ_2 is the second largest eigenvalue of Σ, and $\boldsymbol{\theta}_2$ is the corresponding eigenvector. Similarly, it can be shown that for $i = 1, 2, \ldots, k$, the i-th PC is given by $z_i = \mathbf{x}\boldsymbol{\theta}_i$ where $\boldsymbol{\theta}_i$ is an eigenvector of $\boldsymbol{\Sigma}$ corresponding to its i-th largest eigenvalue λ_i, and $\mathrm{Var}\,(z_i) = \lambda_i$.

3.1.2 Singular Value Decomposition

The singular value decomposition gives $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}'$, where

- $\mathbf{U}(n \times n)$ and $\mathbf{V}(k \times k)$ are orthogonal matrices hence $\mathbf{U}'\mathbf{U} = \mathbf{I}_n$ and $\mathbf{V}'\mathbf{V} = \mathbf{I}_k$;

- $\mathbf{D}$ is an $n \times k$ rectangular matrix. When $n < k$, $\mathbf{D}$ can be jointly formed by an $n \times n$ diagonal matrix and an $n \times (k - n)$ zero matrix.

The singular value decomposition gives

$$\mathbf{X}'\mathbf{X} = (\mathbf{UDV}')'\mathbf{UDV}' = \mathbf{VD}'\mathbf{U}'\mathbf{UDV}' = \mathbf{V}\mathbf{\Lambda}\mathbf{V}', \tag{3.1}$$

where $\mathbf{\Lambda}(k \times k) = \mathbf{D}'\mathbf{D}$. The singular value decomposition of $\mathbf{X}$ gives $\mathbf{V}$ whose columns are eigenvectors of $\mathbf{X}'\mathbf{X}$ and $\mathbf{\Lambda}$ whose diagonal entries are eigenvalues of $\mathbf{X}'\mathbf{X}$ in a descending order. Similarly the eigenvectors of $\mathbf{XX}'$ make up the columns of $\mathbf{U}$. The diagonal elements of $\mathbf{D}$ are the square roots of the eigenvalues of $\mathbf{X}'\mathbf{X}$ or $\mathbf{XX}'$. The singular value decomposition of $\mathbf{X}$ provides a convenient way to find the eigenvalues and the eigenvectors of $\mathbf{X}'\mathbf{X}$ without actually calculating the product.

3.1.3 Principal Components Regression

The bilinear principal components regression model for a single variable with centered data $\mathbf{X}_c$ and $\mathbf{y}_c$ can be written as

$$\begin{aligned} \mathbf{T} &= \mathbf{X}_c\mathbf{V}_a, \\ \mathbf{X}_c &= \mathbf{TP}' + \mathbf{E}, \\ \mathbf{y}_c &= \mathbf{Tq}' + \mathbf{f}, \end{aligned} \tag{3.2}$$

where the (i, j)-th element of $\mathbf{T}$ $(n \times a)$ is the value (score) on the j-th principal component for the i-th observation, and a is the number of principal components. The loadings of the centred response variable denotes as $\mathbf{q}$ $(1 \times a)$. The loading matrix $\mathbf{P}$ $(k \times a)$ and $\mathbf{V}_a$ are identical.

After the number of principal components a is chosen, the weight matrix $\mathbf{P}$ $(k \times a)$ is truncated from a full weight matrix $\mathbf{V}$ $(k \times k)$, corresponding to the a principal components. We calculate the eigenvectors and the square root of eigenvalues of centred explanatory variables, $\mathbf{P}$ and $\mathbf{D}$ $(n \times k)$, from the singular value decomposition (See Section 3.1.2). The eigenvalues $\mathbf{\Lambda} = \mathbf{D}'\mathbf{D} = \begin{pmatrix} \lambda_1 & 0 & \cdots \\ 0 & \ddots & 0 \\ \vdots & 0 & \lambda_a \end{pmatrix}$. The

orthogonality properties of the loadings and the scores give $\mathbf{P}'\mathbf{P} = \mathbf{I}$ and $\mathbf{T}'\mathbf{T} = \mathbf{\Lambda}$. The maximisation criterion for principal components regression corresponds to

$$\begin{aligned} &\text{maximise} && \text{Var}\,(\mathbf{X}_c\mathbf{v}_i) \\ &\text{subject to} && \mathbf{v}_i'\mathbf{v}_i = 1 \text{ and } \text{Cov}\,(\mathbf{X}_c\mathbf{v}_i, \mathbf{X}_c\mathbf{v}_j) = 0, \text{ for } i \neq j. \end{aligned}$$

We solve $\mathbf{X}_c'\mathbf{X}_c\mathbf{V} = \mathbf{V}\mathbf{\Lambda}$ ($\mathbf{X}_c'\mathbf{X}_c\mathbf{v}_i = \lambda_i\mathbf{v}_i$) to obtain $\mathbf{V}$, hence the i-th column of $\mathbf{V}$ is the i-th eigenvector of $\mathbf{X}_c'\mathbf{X}_c$ associated with the eigenvalue λ_i. In this chapter, we use $\tilde{\boldsymbol{\beta}}$ to denote the estimated regression coefficients in the ordinary least squares regression, and use $\hat{\mathbf{q}}$ and $\hat{\boldsymbol{\beta}}$ to denote the ordinary least squares estimates of y-loadings and the estimated regression coefficients in the principal components regression. Since the columns of $\mathbf{T}$ are orthogonal, the ordinary least estimator of y-loadings

$$\begin{aligned} \hat{\mathbf{q}} &= (\mathbf{T}'\mathbf{T})^{-1}\mathbf{T}'\mathbf{y}_c \\ &= ((\mathbf{X}_c\mathbf{V}_a)'\mathbf{X}_c\mathbf{V}_a)^{-1}\mathbf{T}'\mathbf{y}_c = (\mathbf{V}_a'\mathbf{X}_c'\mathbf{X}_c\mathbf{V}_a)^{-1}\mathbf{T}'\mathbf{y}_c \\ &= (\mathbf{V}_a'\mathbf{V}_a\mathbf{D}'\mathbf{D}\mathbf{V}_a'\mathbf{V}_a)^{-1}\mathbf{T}'\mathbf{y}_c \\ &= (\mathbf{D}'\mathbf{D})^{-1}\mathbf{T}'\mathbf{y}_c = \mathbf{\Lambda}^{-1}\mathbf{T}'\mathbf{y}_c. \end{aligned} \tag{3.3}$$

Under the centred regression model $\mathbf{y}_c = \mathbf{X}_c\boldsymbol{\beta} + \boldsymbol{\epsilon}_c$, the principal components regression coefficient estimate $\hat{\boldsymbol{\beta}}$ and its variance can be written as

$$\begin{aligned} \hat{\boldsymbol{\beta}} &= \mathbf{V}_a\hat{\mathbf{q}} \\ &= \mathbf{V}_a(\mathbf{T}'\mathbf{T})^{-1}\mathbf{T}'\mathbf{y}_c = \mathbf{V}_a\mathbf{\Lambda}^{-1}\mathbf{V}_a'\mathbf{X}_c'\mathbf{y}_c \\ &= \sum_{i=1}^{a}\lambda_i^{-1}v_iv_i'\mathbf{X}_c'\mathbf{y}_c; \end{aligned} \tag{3.4}$$

$$\begin{aligned} \text{Var}\,(\hat{\boldsymbol{\beta}}) &= \mathbf{V}_a(\mathbf{T}'\mathbf{T})^{-1}\mathbf{T}'\text{Var}\,(\mathbf{y}_c)(\mathbf{V}_a(\mathbf{T}'\mathbf{T})^{-1}\mathbf{T}')' \\ &= \mathbf{V}_a(\mathbf{T}'\mathbf{T})^{-1}\mathbf{T}'\sigma_\epsilon^2\mathbf{I}_n(\mathbf{V}_a(\mathbf{T}'\mathbf{T})^{-1}\mathbf{T}')' \\ &= \sigma_\epsilon^2\mathbf{V}_a(\mathbf{T}'\mathbf{T})^{-1}\mathbf{T}'\mathbf{T}(\mathbf{T}'\mathbf{T})^{-1}\mathbf{V}_a' \\ &= \sigma_\epsilon^2\mathbf{V}_a(\mathbf{T}'\mathbf{T})^{-1}\mathbf{V}_a' \qquad (= \sigma_\epsilon^2\hat{\mathbf{P}}(\mathbf{T}'\mathbf{T})^{-1}\hat{\mathbf{P}}') \end{aligned} \tag{3.5}$$

$$= \sigma_\epsilon^2\mathbf{V}_a\mathbf{\Lambda}^{-1}\mathbf{V}_a' = \sigma_\epsilon^2\sum_{i=1}^{a}\lambda_i^{-1}v_iv_i'. \tag{3.6}$$

If multi-collinearity exists, principal components with very small variances are omitted because they have small eigenvalues resulting in very large λ_i^{-1}. The

ordinary least squares estimator is unbiased, i.e. $\mathrm{E}(\tilde{\boldsymbol{\beta}}) = \boldsymbol{\beta}$, but the regression coefficient estimates in principal components regression is not unbiased, because

$$\hat{\boldsymbol{\beta}} = \tilde{\boldsymbol{\beta}} - \sum_{i=a+1}^{k} \lambda_i^{-1} v_i v_i' \mathbf{X}_c' \mathbf{y}_c,$$

but $\hat{\boldsymbol{\beta}}$ is more stable than $\tilde{\boldsymbol{\beta}}$.

3.2 Principal Components Regression Prediction Uncertainty

The linear models for a single response variable principal components regression can be expressed as

$$\dot{\mathbf{y}}_c = \beta_0 + \dot{\mathbf{X}}_c \boldsymbol{\beta} + \boldsymbol{\epsilon}_c,$$
$$\dot{\mathbf{y}}_p = \beta_0 + \dot{\mathbf{X}}_p \boldsymbol{\beta} + \boldsymbol{\epsilon}_p,$$

where β_0 and $\boldsymbol{\beta}$ $(k \times 1)$ are regression coefficients, $\dot{\mathbf{y}}_c$ and $\dot{\mathbf{y}}_p$ are calibration and prediction response variables. $\dot{\mathbf{X}}_c$ $(n \times k)$ and $\dot{\mathbf{X}}_p$ $(n_p \times k)$ are calibration and prediction explanatory variables matrices. Let j index the observations: in the calibration set $j = 1, \ldots, n$, while in the prediction set $j = 1, \ldots, n_p$. $\boldsymbol{\epsilon}_c$ $(n \times 1)$ and $\boldsymbol{\epsilon}_p$ $(n_p \times 1)$ are the error terms in the calibration and the prediction sets. The term error $\mathbf{f}$ in Equation (3.2) of the bilinear model equals to the sum of the bias and the regression error from the last step of principal components regression, which is an ordinary least squares regression.

Two traditional approaches, the empirical estimates of prediction mean squared error and the ordinary least squares type prediction mean squared error, are used in the quantification of principal components regression prediction uncertainty.

3.2.1 Simple Empirical Estimates: Root Mean Squared Error of Prediction (RMSEP) and Root Mean Squared Error of Cross-validation (RMSECV)

A simple estimate of prediction uncertainty is provided by the root mean squared error of prediction (RMSEP) in Section 1.1. It empirically estimates the combination of the variance and the bias, but it is an average uncertainty.

An alternative is the root mean squared prediction error of cross-validation (RMSECV). Leave-one-out cross-validation is a standard tool to obtain nearly an unbiased estimator of prediction error. Each observation has been left out and predicted once, then the root mean squared prediction error of cross-validation is calculated using these predictions and predicted values.

$$\text{RMSECV} = \sqrt{\text{MSECV}} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(\dot{y}_{c_j} - \hat{\alpha}_{cv_j} - \mathbf{x}_{c_j}\hat{\boldsymbol{\beta}}_{cv_j})^2}, \tag{3.7}$$

where $\hat{\alpha}_{cv_j}$ and $\hat{\boldsymbol{\beta}}_{cv_j}$ are regression coefficient estimates of a reduced dataset that does not include the j-th observation.

Both RMSEP and RMSECV are meaningful for principal components regression as it is a biased regression method. The two empirical estimates consider the variation in the regression and the bias as a whole.

3.2.2 Ordinary Least Squares Type Prediction Mean Squared Error

Since principal components regression is based on ordinary least squares regression, a direct thought following this is to use Equation (3.5) $\text{Var}\,(\hat{\boldsymbol{\beta}}) = \sigma_\epsilon^2\hat{\mathbf{P}}(\mathbf{T}'\mathbf{T})^{-1}\hat{\mathbf{P}}'$ as the expression of $\text{Var}\,(\hat{\boldsymbol{\beta}})$. The ordinary least squares type prediction mean squared error can be written according to $\hat{\dot{y}}_p = \bar{\dot{y}} + \mathbf{x}_p\hat{\boldsymbol{\beta}}$,

$$\text{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \frac{\sigma_\epsilon^2}{n} + \sigma_\epsilon^2\mathbf{x}_p\hat{\mathbf{P}}(\mathbf{T}'\mathbf{T})^{-1}\hat{\mathbf{P}}'\mathbf{x}_p' + \sigma_\epsilon^2, \tag{3.8}$$

where the centred predictors $\mathbf{x}_p = \dot{\mathbf{x}}_p - \bar{\dot{\mathbf{x}}}$ and the leverage $h = \mathbf{x}_p\hat{\mathbf{P}}(\mathbf{T}'\mathbf{T})^{-1}\hat{\mathbf{P}}'\mathbf{x}_p' = \mathbf{t}_p(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_p'$. It suggests a linear relationship between the prediction mean squared

error and the leverage. The regression error variance σ_ϵ^2 can be estimated in two ways.

- An estimate from the calibration set:

$$\hat{\sigma}_{\epsilon_c}^2 = \text{MSEC} = \frac{1}{n-a-1}\sum_{j=1}^{n}(\dot{y}_{c_j} - \hat{\alpha} - \mathbf{x}_{c_j}\hat{\boldsymbol{\beta}})^2, \tag{3.9}$$

  where a is the number of principal components. The divisor $n-a-1$ uses the number of factors instead of the true number of explanatory variables involved, which is unknown, so it underestimates the regression error variance.

- An estimate using the tuning set:

  The idea of estimating the regression error variance from the tuning set is the same as shown in Section 1.1 where the regression error variance is estimated from a tuning set. Similarly to Equation (1.9), the estimated regression error variance can be calculated as

$$\hat{\sigma}_{\epsilon_t}^2 = \frac{\text{MSEP}}{\frac{1}{n_t-1} + \frac{1}{n}\sum_{j=1}^{n_t} h_{t_j} + 1}, \tag{3.10}$$

  where the leverage $h_{t_j} = \mathbf{t}_{t_j}(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}'_{t_j}$. $\mathbf{t}_{t_j}$ denotes the scores of the centred explanatory variables in the tuning set, $\mathbf{t}_{t_j} = (\dot{\mathbf{x}}_{t_j} - \bar{\dot{\mathbf{x}}})\hat{\mathbf{P}}$. $\mathbf{T}$ is the score matrix of centred explanatory variables. Both Equations (3.10) and (1.9) is the ratio of average residual sum of squares and a function of average leverage.

3.3 Principal Components Regression Simulation Study

We use simulation studies to explore the quantification of principal components regression prediction uncertainty. The performance of the ordinary least squares type prediction mean squared error is studied for independent normally distributed explanatory variables in Simulation 3.1. We use Simulation 3.2 to show that the

bias is the source of the missing part in the prediction mean squared error formula. We also think of potential better solutions to prediction uncertainty. We try to measure prediction mean squared error in terms of sample size in Simulation 3.3. Meanwhile, we find the bias and the leverage may be correlated. Simulation 3.4 studies the correlation between the bias and the leverage, and discusses how the correlation affects the measurement of prediction uncertainty .

Principal components regression has an extra step to choose the number of principal components before the calibration is carried out. Here we use leave-one-out cross-validation to decide the number of principal components.

There are N replicates of simulations, each of which consists of a calibration set, a tuning set and a prediction set. It is the same as Case (3) in Simulation 2.1, where the design of different $\dot{\mathbf{X}}$ and different $\dot{\mathbf{y}}$ is being used. Every calibration set contains n observations. The number of observations in the tuning set is denoted as n_t. There are n_p observations in the prediction set, and there are k explanatory variables. A tuning set is simulated in order to estimate the regression error variance and investigate how the estimated regression error variance $\hat{\sigma}_\epsilon^2$ plays a role in the quantification of prediction uncertainty.

Each replicate of principal components regression simulation has a different calibration set and a prediction set since prediction uncertainty depends not only on the leverage but also on the choice of factors, so the procedure to choose the factors among different calibration explanatory variables needs to be considered. The general procedures are written as below. We use a simple structure of explanatory variables in which they are independent and identically normally distributed. The tuning, and prediction sets are simulated from the same distribution of the calibration set because good practice in calibration is to make the training set representative of future samples. Of course an extensive simulation study would need to explore both correlated predictors and the effect of extrapolation, but our purpose here is just to demonstrate some of the properties of the methods investigated using a few simple simulations.

1. The simulations of the calibration set, the tuning set and the prediction set

$\dot{\mathbf{X}}_c$ $(n \times k)$ are generated from independent normal distributions with mean $\mathbf{0}$ and variances $\sigma^2_{c_1}, \ldots, \sigma^2_{c_k}$. Let a be the number of explanatory variables taking over most variations in the response variable. The noise $\boldsymbol{\epsilon}_c$ is simulated from the normal distribution with mean 0 and variance σ^2_ϵ. The observations in the calibration set can be expressed as $\dot{\mathbf{y}}_c = \beta_0 + \dot{\mathbf{X}}_c\boldsymbol{\beta} + \boldsymbol{\epsilon}_c$.

The explanatory variables in the tuning set $\dot{\mathbf{X}}_t$ is simulated as the same as the calibration set, hence $\dot{\mathbf{y}}_t = \beta_0 + \dot{\mathbf{X}}_t\boldsymbol{\beta} + \boldsymbol{\epsilon}_t$. $\boldsymbol{\epsilon}_t$ is the error term in the tuning set, which has the normal distribution with mean 0 and variance σ^2_ϵ.

For the prediction set, taking the j-th prediction observation as an example, the predictor $\dot{\mathbf{x}}_{p_j}$ $(1 \times k)$ is simulated from the same distribution as the calibration. The predictions can be calculated as $\dot{\mathbf{y}}_p = \beta_0 + \dot{\mathbf{X}}_p\boldsymbol{\beta} + \boldsymbol{\epsilon}_p$. The error term $\boldsymbol{\epsilon}_p$ has a normal distribution with mean 0 and variance σ^2_ϵ.

2. The calibration and the prediction

After we find loadings $\hat{\mathbf{P}}$ and $\hat{\mathbf{q}}$ via the singular value decomposition, the scores of a predictor $\mathbf{t}_p = (\dot{\mathbf{x}}_p - \bar{\dot{\mathbf{x}}})\hat{\mathbf{P}}$, and the predicted value can be calculated as $\hat{\dot{y}}_p = \bar{\dot{y}} + \mathbf{t}_p\hat{\mathbf{q}}'$. Since $\frac{1}{n}\mathbf{X}_c'\mathbf{X}_c$ is the sample covariance from $N(0, \boldsymbol{\Sigma}_k)$, $\mathrm{E}\,(\frac{1}{n}\mathbf{X}_c'\mathbf{X}_c) = \boldsymbol{\Sigma}_k$, so $(\mathbf{X}_c'\mathbf{X}_c)^{-1} \approx \frac{1}{n}\boldsymbol{\Sigma}_k^{-1}$. The scores $\mathbf{T}$ correspond to a principal components of $\mathbf{X}_c'\mathbf{X}_c$ and the largest a eigenvalues $\lambda_1, \ldots, \lambda_a$, so $(\mathbf{T}'\mathbf{T})^{-1} \approx \frac{1}{n}\boldsymbol{\Sigma}_a^{-1}$. The leverage of the j-th prediction sample, $h_j = \mathbf{t}_{p_j}(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_{p_j}'$. According to Searle (1997) Page 55, $\mathrm{E}\,\{\mathbf{t}_{p_j}(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_{p_j}'\} = \mathrm{E}\left[\mathrm{tr}\{(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_{p_j}'\mathbf{t}_{p_j}\}\right] = \mathrm{tr}\{(\mathbf{T}'\mathbf{T})^{-1}\mathrm{E}\,(\mathbf{t}_{p_j}'\mathbf{t}_{p_j})\}$. Because $\mathrm{E}\,(\mathbf{t}_{p_j}) = \mathbf{0}$, $\mathrm{E}\,(\mathbf{t}_{p_j}'\mathbf{t}_{p_j}) = \boldsymbol{\Sigma}_a$, thus $\mathrm{E}\,(h_j) \approx \mathrm{tr}(\frac{1}{n}\boldsymbol{\Sigma}_a^{-1} \times \boldsymbol{\Sigma}_a) = \frac{a}{n}$.

3.3.1 PCR Simulation with Noise Free Prediction Samples

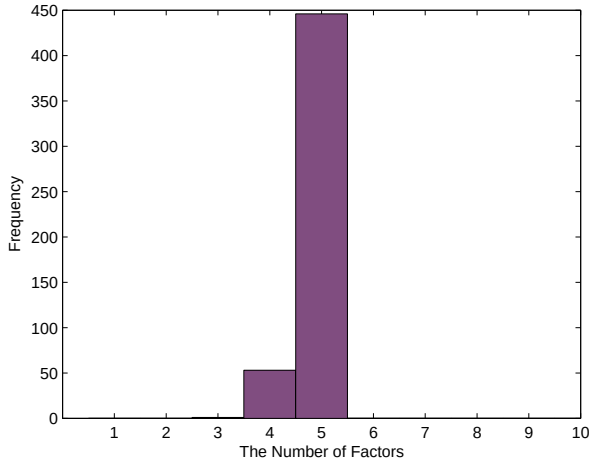
Simulation 3.1. Principal Components Regression Simulation Study

The simulations can be run in a similar way as the ordinary least squares regression simulation. We study two cases and use leave-one-out cross-validation to choose the number of principal components. $k = 50$, $a = 5$, $N = 10000$,

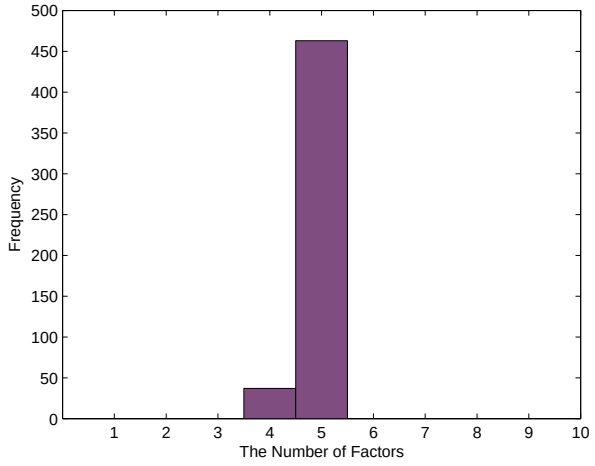
$n = 100$, $n_p = 100$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$, $\beta_0 = 1$, $\boldsymbol{\beta} = (\ 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad \cdots \quad 0\ )$ and

(a) $\boldsymbol{\sigma}_c^2 = \ (10^2 \quad 10^2 \quad 10^2 \quad 10^2 \quad 10^2 \quad 1 \quad \cdots \quad 1\ )$.

(b) $\boldsymbol{\sigma}_c^2 = (\ 10^6 \quad 10^6 \quad 10^6 \quad 10^6 \quad 10^6 \quad 1 \quad \cdots \quad 1\ )$.



(a) Case (a)



(b) Case (b)

Figure 3.1: PCR Histogram: the Number of Principal Components

The histograms of the number of principal components for 500 replicates are shown in Figure 3.1. It can be seen that for Case (a) and (b) the number of principal components can be chosen as 5, which is consistent with the experiment design,

where large variances are allocated to the first five explanatory variables, and suitable regression coefficients enable explanatory variables with large variances to take over most variations in the response variable.

As the error term $\boldsymbol{\epsilon}_p$ in the prediction set is assumed to be zero according to Equation (3.8), the ordinary least squares type prediction mean squared error can be written as

$$\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \sigma_\epsilon^2(\frac{1}{n} + h). \tag{3.11}$$

Average squared prediction error and average leverage are obtained by Definition 2.1 Chi-square binning method, where we set the number of bins to be 20.

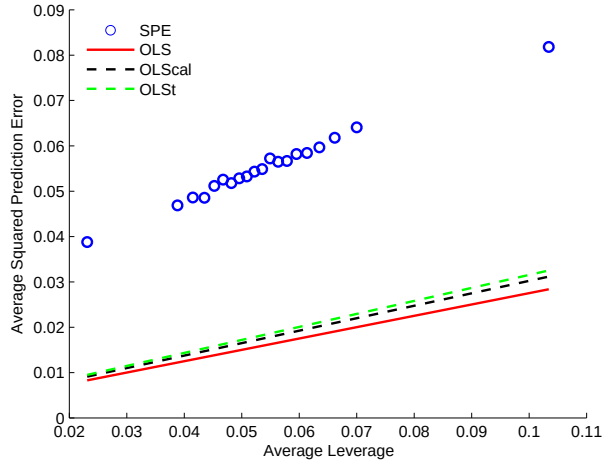
In Figure 3.2, the blue points (SPE) present average squared prediction error against average leverage. The red line (OLS) is the ordinary least squares type prediction mean squared error with $\sigma_\epsilon^2 = 0.25$. The black dash line (OLScal) gives the ordinary least squares type prediction mean squared error with the estimated regression error variance $\hat{\sigma}_{\epsilon_c}^2$ from the calibration set, and the green dash line (OLSt) stands for the ordinary least squares type prediction mean squared error with the regression error variance estimate $\hat{\sigma}_{\epsilon_t}^2$ from the tuning set.

The gap between the blue point line and the red line in Figure 3.2(a) is attributed to unselected components. It also tells that the regression error variance estimate from the tuning set seems not have an effect that compensates the omission of unused explanatory variables as it claims to, which indicates that there might be something missing from the ordinary least squares type expression. We will discuss it in Section 3.3.2.

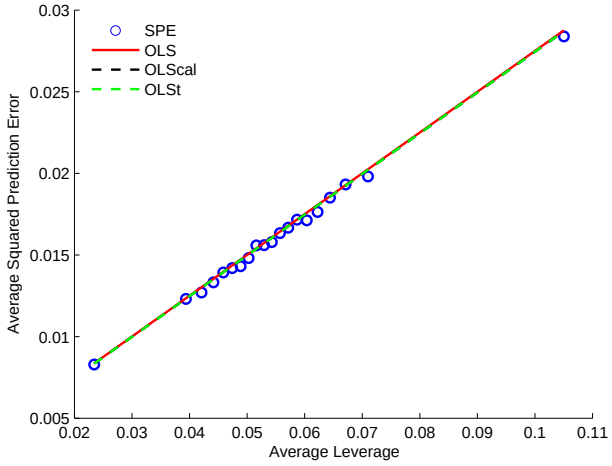
Figure 3.2(b) shows that in Case (b) the principal components regression is equivalent to the ordinary least squares regression when $k = 5$, because the first five explanatory variables have very large variances.

3.3.2 Bias and PCR Prediction Uncertainty

Simulation 3.1 has shown the ordinary least squares type prediction mean squared error only accounts for a part of prediction uncertainty. It is why there is a gap between average squared prediction error against average leverage line and the



(a) Case (a)



(b) Case (b)

Figure 3.2: PCR Average Squared Prediction Error versus Average Leverage, i.i.d. standard normally distributed predictors, $\boldsymbol{\epsilon}_p = \mathbf{0}$. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$ against average leverage $h = \mathbf{t}_p(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_p'$. OLS: the ordinary least squares type prediction mean squared error using the true regression variance $\mathrm{E}\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} = \sigma_\epsilon^2(\frac{1}{n} + h)$. OLScal: the ordinary least squares type prediction mean squared error using the estimated regression variance from the calibration set $\mathrm{E}\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} = \hat{\sigma}_{\epsilon_c}^2(\frac{1}{n} + h)$. OLSt: the ordinary least squares type prediction mean squared error using the estimated regression variance from the tuning set $\mathrm{E}\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} = \hat{\sigma}_{\epsilon_t}^2(\frac{1}{n} + h)$.

ordinary least squares type expression line in Figure 3.2, where $\hat{\sigma}^2_{\epsilon_t}$ seems not to compensate for the omission of unselected explanatory variables. We are going to carry out the investigation of the bias since the principal components regression theory suggests the bias contributing to parts of the prediction mean squared error. We will demonstrate expected squared bias is the missing part of prediction mean squared error in a simple case, and a simulation result will be used to verify this.

Simulation 3.2. Ordinary Least Squares type Principal Components Prediction Mean Squared Error Adjustment

Following the numerical experiment of Simulation 3.1 Case (a) where $k = 50$, $a = 5$ and $\boldsymbol{\beta} = (\ 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad \cdots \quad 0\ )$, under the full model, a prediction, its fitted value and the residual can be written as below.

$$\begin{aligned}
\dot{y}_p &= \alpha + \mathbf{x}_p\boldsymbol{\beta} + \epsilon_p = \alpha + \sum_{l=1}^{a} t_{p_l}\mathbf{v}_l'\boldsymbol{\beta} + \sum_{l=a+1}^{k} t_{p_l}\mathbf{v}_l'\boldsymbol{\beta} + \epsilon_p. \\
\hat{\dot{y}}_p &= \hat{\alpha} + \mathbf{t}_p\hat{\mathbf{q}}' = \hat{\alpha} + \sum_{l=1}^{a} t_{p_l}\mathbf{v}_l'\hat{\boldsymbol{\beta}}. \\
\dot{y}_p - \hat{\dot{y}}_p &= \alpha - \hat{\alpha} + \sum_{l=1}^{a} t_{p_l}\mathbf{v}_l'(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}) + \sum_{l=a+1}^{k} t_{p_l}\mathbf{v}_l'\boldsymbol{\beta} + \epsilon_p \\
&= \alpha - \hat{\alpha} + \mathbf{t}_p(\mathbf{q}' - \hat{\mathbf{q}}') + \sum_{l=a+1}^{k} t_{p_l}\mathbf{v}_l'\boldsymbol{\beta} + \epsilon_p.
\end{aligned}$$

Hence, the prediction mean squared error can be expressed as

$$\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \frac{\sigma^2_\epsilon}{n} + \sigma^2_\epsilon\mathbf{t}_p(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_p' + \mathrm{E}\,(\sum_{l=a+1}^{k} t_{p_l}\mathbf{v}_l'\boldsymbol{\beta})^2 + \sigma^2_\epsilon, \tag{3.12}$$

where the bias $= \sum_{l=a+1}^{k} t_{p_l} \mathbf{v}_l' \boldsymbol{\beta}$, and it can be expanded as

$$
\begin{aligned}
& \sum_{l=a+1}^{k} \mathbf{t}_{p_l} \mathbf{v}_{l}' \boldsymbol{\beta} = \sum_{l=a+1}^{k} \mathbf{x}_p \mathbf{v}_{l} \mathbf{v}_{l}' \boldsymbol{\beta} \\
= & \sum_{l=a+1}^{k} (\sum_{i=1}^{k} x_{p_i} v_{li})(\sum_{i=1}^{k} v_{li} \beta_i) = \sum_{l=6}^{50} \{(\sum_{i=1}^{50} x_{p_i} v_{li})(\sum_{i=1}^{5} v_{li})\} \\
= & \sum_{l=6}^{50} \big(\sum_{i=1}^{50} x_{p_i} v_{li} v_{l1} + \sum_{i=1}^{50} x_{p_i} v_{li} v_{l2} + \sum_{i=1}^{50} x_{p_i} v_{li} v_{l3} + \sum_{i=1}^{50} x_{p_i} v_{li} v_{l4} + \sum_{i=1}^{50} x_{p_i} v_{li} v_{l5} \big) \\
= & \sum_{l=6}^{50} \big(x_{p_1} v_{l1}^2 + x_{p_2} v_{l2} v_{l1} + x_{p_3} v_{l3} v_{l1} + x_{p_4} v_{l4} v_{l1} + x_{p_5} v_{l5} v_{l1} + \cdots + x_{p_{50}} v_{l50} v_{l1} \\
& + x_{p_1} v_{l1} v_{l2} + x_{p_2} v_{l2}^2 + x_{p_3} v_{l3} v_{l2} + x_{p_4} v_{l4} v_{l2} + x_{p_5} v_{l5} v_{l2} + \cdots + x_{p_{50}} v_{l50} v_{l2} \\
& + x_{p_1} v_{l1} v_{l3} + x_{p_2} v_{l2} v_{l3} + x_{p_3} v_{l3}^2 + x_{p_4} v_{l4} v_{l3} + x_{p_5} v_{l5} v_{l3} + \cdots + x_{p_{50}} v_{l50} v_{l3} \\
& + x_{p_1} v_{l1} v_{l4} + x_{p_2} v_{l2} v_{l4} + x_{p_3} v_{l3} v_{l4} + x_{p_4} v_{l4}^2 + x_{p_5} v_{l5} v_{l4} + \cdots + x_{p_{50}} v_{l50} v_{l4} \\
& + x_{p_1} v_{l1} v_{l5} + x_{p_2} v_{l2} v_{l5} + x_{p_3} v_{l3} v_{l5} + x_{p_4} v_{l4} v_{l5} + x_{p_5} v_{l5}^2 \cdots + x_{p_{50}} v_{l50} v_{l5} \big).
\end{aligned}
$$

And then, we could estimate the expected squared bias, and then replace $\mathrm{E}\,(\sum_{l=a+1}^{k} t_{p_l} \mathbf{v}_l' \boldsymbol{\beta})^2$ with the estimated expected squared bias.

The expected squared bias can be written as

$$
\begin{aligned}
& \mathrm{E}\,(\sum_{l=A+1}^{k} \mathbf{t}_{p_l}\mathbf{v}_l'\boldsymbol{\beta})^2 \\
= & \sum_{l=6}^{50}\big\{x_{p_1}^2\mathrm{E}\,(v_{l1}^4)+x_{p_2}^2\mathrm{E}\,(v_{l2}^2)\mathrm{E}\,(v_{l1}^2)+x_{p_3}^2\mathrm{E}\,(v_{l3}^2)\mathrm{E}\,(v_{l1}^2)+x_{p_4}^2\mathrm{E}\,(v_{l4}^2)\mathrm{E}\,(v_{l1}^2)+\cdots+x_{p_{50}}^2\mathrm{E}\,(v_{l50}^2)\mathrm{E}\,(v_{l1}^2) \\
& +2x_{p_1}x_{p_2}\mathrm{E}\,(v_{l1}^2)\mathrm{E}\,(v_{l2}^2)+2x_{p_1}x_{p_3}\mathrm{E}\,(v_{l1}^2)\mathrm{E}\,(v_{l3}^2)+2x_{p_1}x_{p_4}\mathrm{E}\,(v_{l1}^2)\mathrm{E}\,(v_{l4}^2)+2x_{p_1}x_{p_5}\mathrm{E}\,(v_{l1}^2)\mathrm{E}\,(v_{l5}^2) \\
& +2x_{p_2}x_{p_3}\mathrm{E}\,(v_{l2}^2)\mathrm{E}\,(v_{l3}^2)+2x_{p_2}x_{p_4}\mathrm{E}\,(v_{l2}^2)\mathrm{E}\,(v_{l4}^2)+2x_{p_2}x_{p_5}\mathrm{E}\,(v_{l2}^2)\mathrm{E}\,(v_{l5}^2) \\
& +2x_{p_3}x_{p_4}\mathrm{E}\,(v_{l3}^2)\mathrm{E}\,(v_{l4}^2)+2x_{p_3}x_{p_5}\mathrm{E}\,(v_{l3}^2)\mathrm{E}\,(v_{l5}^2) \\
& +2x_{p_4}x_{p_5}\mathrm{E}\,(v_{l4}^2)\mathrm{E}\,(v_{l5}^2)\big\} \\
+ & 2\sum_{\substack{l=6\\ l<g}}^{50}\big\{x_{p_1}^2\mathrm{E}\,(v_{l1}^2)\mathrm{E}\,(v_{g1}^2)+x_{p_1}x_{p_2}\mathrm{E}\,(v_{l1}^2)\mathrm{E}\,(v_{g2}^2)+x_{p_1}x_{p_3}\mathrm{E}\,(v_{l1}^2)\mathrm{E}\,(v_{g3}^2)+\cdots+x_{p_1}x_{p_5}\mathrm{E}\,(v_{l1}^2)\mathrm{E}\,(v_{g5}^2) \\
& +x_{p_1}x_{p_2}\mathrm{E}\,(v_{l2}^2)\mathrm{E}\,(v_{g1}^2)+x_{p_2}^2\mathrm{E}\,(v_{l2}^2)\mathrm{E}\,(v_{g2}^2)+x_{p_2}x_{p_3}\mathrm{E}\,(v_{l2}^2)\mathrm{E}\,(v_{g3}^2)+\cdots+x_{p_2}x_{p_5}\mathrm{E}\,(v_{l2}^2)\mathrm{E}\,(v_{g5}^2) \\
& +x_{p_1}x_{p_3}\mathrm{E}\,(v_{l3}^2)\mathrm{E}\,(v_{g1}^2)+x_{p_2}x_{p_3}\mathrm{E}\,(v_{l2}^2)\mathrm{E}\,(v_{g3}^2)+x_{p_3}^2\mathrm{E}\,(v_{l3}^2)\mathrm{E}\,(v_{g3}^2)+\cdots+x_{p_3}x_{p_5}\mathrm{E}\,(v_{l3}^2)\mathrm{E}\,(v_{g5}^2) \\
& +x_{p_1}x_{p_4}\mathrm{E}\,(v_{l4}^2)\mathrm{E}\,(v_{g1}^2)+x_{p_2}x_{p_4}\mathrm{E}\,(v_{l4}^2)\mathrm{E}\,(v_{g2}^2)+x_{p_3}x_{p_4}\mathrm{E}\,(v_{l4}^2)\mathrm{E}\,(v_{g3}^2)+\cdots+x_{p_4}x_{p_5}\mathrm{E}\,(v_{l4}^2)\mathrm{E}\,(v_{g5}^2) \\
& +x_{p_1}x_{p_5}\mathrm{E}\,(v_{l5}^2)\mathrm{E}\,(v_{g1}^2)+x_{p_2}x_{p_5}\mathrm{E}\,(v_{l5}^2)\mathrm{E}\,(v_{g2}^2)+x_{p_4}x_{p_5}\mathrm{E}\,(v_{l5}^2)\mathrm{E}\,(v_{g4}^2)+\cdots+x_{p_5}^2\mathrm{E}\,(v_{l5}^2)\mathrm{E}\,(v_{g5}^2)\big\}.
\end{aligned}
$$

Hence, the estimated expected squared bias can be calculated if we have the expectations of squared scores $\mathrm{E}(v^2)$. We run simulations and use the averages of squared scores as the expected squared scores. The simulation is planned as follows.

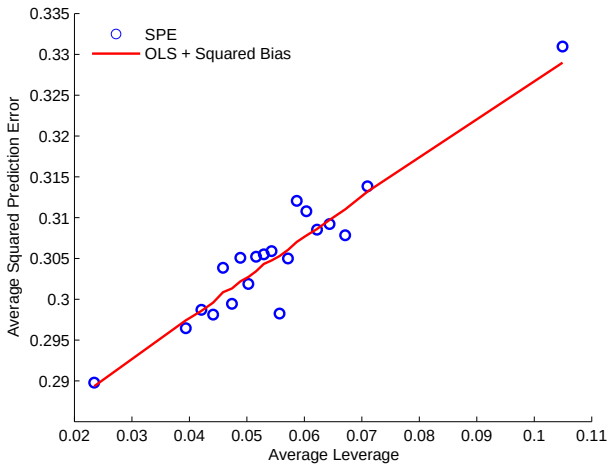


Figure 3.3: PCR Average Squared Prediction Error versus Average Leverage, to verify the missing part of the ordinary least squares type prediction mean squared error is squared bias, $\boldsymbol{\epsilon}_p \neq \mathbf{0}$. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$ against average leverage $h = \mathbf{t}_p(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_p'$. OLS + Squared Bias: the sum of the ordinary least squares type prediction mean squared error and the squared bias $\mathrm{E}\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} = \sigma_\epsilon^2(\frac{1}{n} + h + 1) + \mathrm{E}(\mathrm{bias}^2)$, Equation (3.13).

The simulation is run the same as Simulation 3.1 Case (a), with the addition that the error term $\boldsymbol{\epsilon}_p$ is included in the prediction set, and we assume that $\sigma_\epsilon^2 = 0.25$. To keep it simple we use σ_ϵ^2 rather than its estimate from the calibration set or the tuning set in the ordinary squares type prediction mean squared error formula. In Figure 3.3, the blue point (SPE) denotes average squared prediction error against average leverage. The red line (OLS + Squared Bias) is plotted by the sum of average ordinary least squares type prediction mean squared error and average squared bias against average leverage. The red line fits the blue points, which verifies that it is the expected squared bias missing from the ordinary least squares type prediction mean squared error. Therefore, the principal components

prediction mean squared error can be written as

$$\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \frac{\sigma_\epsilon^2}{n} + \sigma_\epsilon^2 h + \mathrm{E}\left(\mathrm{bias}^2\right) + \sigma_\epsilon^2. \tag{3.13}$$

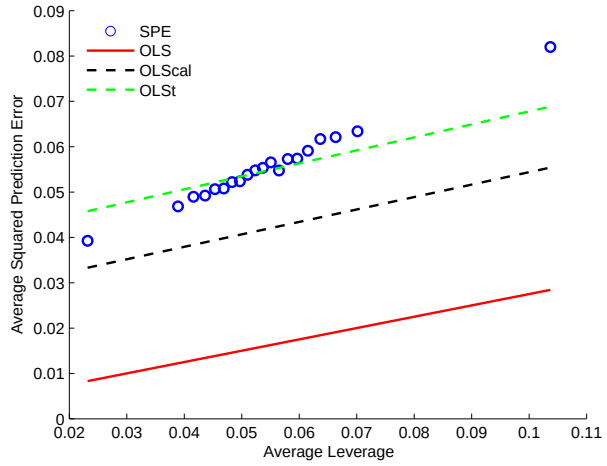
The key parts of $\mathrm{E}\left(\mathrm{bias}^2\right)$ include the calculations using the variance of the loadings, i.e. the eigenvectors of $\mathbf{X}_c'\mathbf{X}_c$. Jolliffe (2002) gives the probability distributions for sample principal components, but the asymptotic results do not provide good approximations for the variance of these eigenvector elements under a moderate sample size, so it would be unrealistic to have a mathematical solution directly derived from Equation (3.13). Before starting the numerical experiment, we obtain the eigenvector matrix $\mathbf{V}$ from the singular value decomposition of the centred explanatory variable matrix repeatedly 10000 times, for which the variance of all elements in $\mathbf{V}$ is calculated. Hence, the expected values relevant to the elements in the eigenvectors in the expected squared bias formula can be directly used from the empirical variance estimate.

Simulation 3.2 shows the expected squared bias is the missing part of the ordinary squares type prediction mean squared error formula theoretically. In practice, how can we estimate it? We will investigate two situations: (1) when the error term is excluded from the prediction sample; (2) when the error term is included in prediction samples.

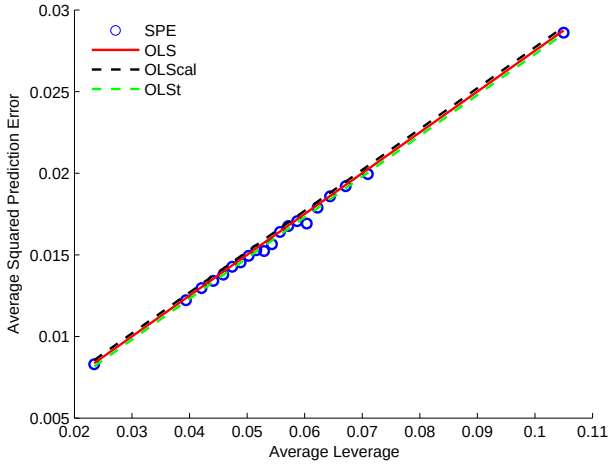
For the case where the error term is not included in the prediction sample, the ordinary least squares type prediction mean squared error formula used in Simulation 3.1, Equation (3.11) can be improved as

$$\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \frac{\sigma_\epsilon^2}{n} + \sigma_\epsilon^2 h + \hat{\mathrm{E}}\left(\mathrm{bias}^2\right), \tag{3.14}$$

where $\hat{\mathrm{E}}\left(\mathrm{bias}^2\right) = \hat{\sigma}_\epsilon^2 - 0.25$. $\hat{\sigma}_\epsilon^2$ contains not only the variation about the ordinary least square regression, the last step of principal components regression, abut also the variation about using the factors. Although ϵ_p is assumed to be zero, the variation about using the factors does exist, and it is the part of $\hat{\sigma}_\epsilon^2$ that does not go into the regression error variance 0.25. To reproduce Figure 3.2 according to Equation (3.14), we presents the results in Figure 3.4. In Figure 3.4(a) the green line (OLSt) crosses the middle of blue points, indicating that the adjusted ordinary



(a) Case (a)



(b) Case (b)

Figure 3.4: PCR Average Squared Prediction Error versus Average Leverage, i.i.d. standard normally distributed predictors, $\boldsymbol{\epsilon}_p = \mathbf{0}$, adjusted ordinary least squares type prediction mean squared error. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$ against average leverage $h = \mathbf{t}_p(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_p'$. OLS: the ordinary least squares type prediction mean squared error using the true regression variance $\mathrm{E}\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} = \sigma_\epsilon^2(\frac{1}{n} + h)$. OLScal: the ordinary least squares type prediction mean squared error using the estimated regression variance from the calibration set $\mathrm{E}\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} = \hat{\sigma}_{\epsilon_c}^2(\frac{1}{n} + h)$. OLSt: the ordinary least squares type prediction mean squared error using the estimated regression variance from the tuning set $\mathrm{E}\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} = \hat{\sigma}_{\epsilon_t}^2(\frac{1}{n} + h)$.

least squares type prediction mean squared error with $\hat{\sigma}^2_{\epsilon_t}$ is right on average. The black line (OLScal) shows that the adjusted ordinary least squares type prediction mean squared error with $\hat{\sigma}^2_{\epsilon_c}$ underestimates prediction uncertainty.

On the other hand, we run Simulation 3.1, but assume $\boldsymbol{\epsilon}_p \neq \mathbf{0}$. The results are shown in Figure 3.5. The prediction mean squared error, Equation (3.13), can be transformed into the adjusted ordinary least squares type prediction mean squared error for principal components regression,

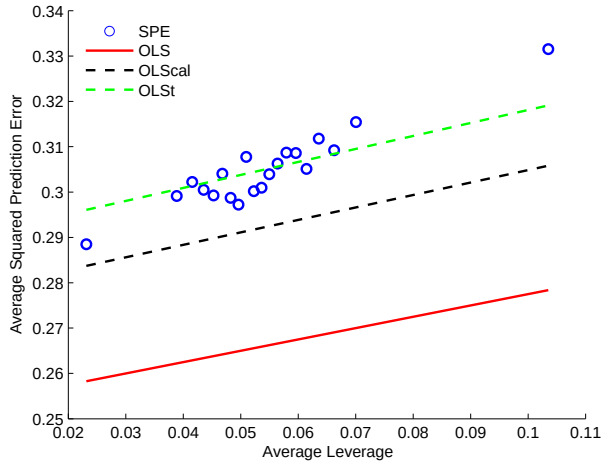
$$\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \frac{\hat{\sigma}^2_\epsilon}{n} + \hat{\sigma}^2_\epsilon h + \hat{\sigma}^2_\epsilon,$$

where $\hat{\sigma}^2_\epsilon$ is the general form of $\hat{\sigma}^2_{\epsilon_c}$ and $\hat{\sigma}^2_{\epsilon_t}$. It contains the information for the variation about the regression and the expected squared bias.

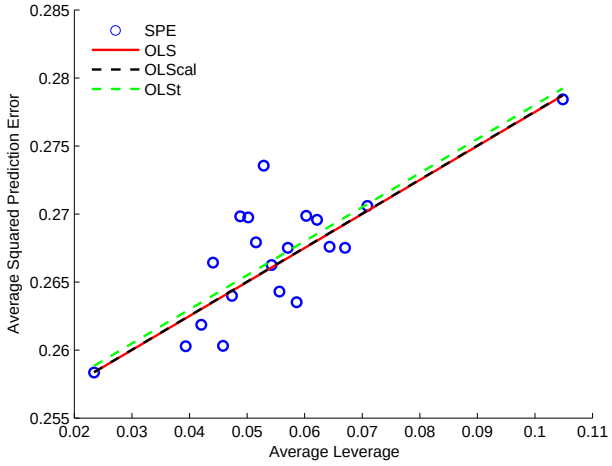
In Figure 3.5, the blue points (SPE) are more noisy compared to those in Figure 3.4 due to $\boldsymbol{\epsilon}_p \neq \mathbf{0}$. In Figure 3.5(a), the red line (OLS) is below the blue point line. It again shows there exist a bias between the squared prediction error and the ordinary least squares type prediction mean squared error if we use $\sigma^2_\epsilon = 0.25$. The black dash line (OLScal) looks close to the blue point line as the estimated regression error variance $\hat{\sigma}^2_{\epsilon_c}$ is calculated from the calibration set, it makes up the omission of unused components to some extend. The green dash line (OLSt) seems to fit the blue points nicely, which suggests the regression error variance estimate $\hat{\sigma}^2_{\epsilon_t}$ compensates the omission of unused components in the ordinary least squares type prediction mean squared error.

Although the blue point lines in Figure 3.5(b) are noisy, the red, the black, and the green lines overlap, delivering the information that the principal components regression in Case (b) is equivalent to the ordinary least squares regression.

It has been shown in this section that the adjusted ordinary least squares type principal components prediction mean squared error works well on average with the estimated regression error variance from the tuning set. Partial least squares regression is similar to principal components regression in terms of the bias, where the bias is made up of the linear combinations of explanatory variables, hence the adjusted ordinary least squares type prediction mean squared error theory also applies.



(a) Case (a)



(b) Case (b)

Figure 3.5: PCR Average Squared Prediction Error versus Average Leverage, i.i.d. standard normally distributed predictors, $\boldsymbol{\epsilon}_p \neq \mathbf{0}$, adjusted ordinary least squares type prediction mean squared error. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$ against average leverage $h = \mathbf{t}_p(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_p'$. OLS: the ordinary least squares type prediction mean squared error using the true regression variance $\mathrm{E}\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} = \sigma_\epsilon^2(\frac{1}{n} + h + 1)$. OLScal: the ordinary least squares type prediction mean squared error using the estimated regression variance from the calibration set $\mathrm{E}\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} = \hat{\sigma}_{\epsilon_c}^2(\frac{1}{n} + h + 1)$. OLSt: the ordinary least squares type prediction mean squared error using the estimated regression variance from the tuning set $\mathrm{E}\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} = \hat{\sigma}_{\epsilon_t}^2(\frac{1}{n} + h + 1)$.

3.3.3 Sample Size and PCR Prediction Uncertainty

There is a $\frac{1}{n}$ term in the prediction mean squared error formula (See Equation (3.13)). The leverage and the bias are also connected to $\frac{1}{n}$, hence an alternative approach to study the relationship between prediction mean squared error and leverage under different sample size is investigated. We would like to explore the relationship among sample size, intercept, and slope in the average squared prediction error against average leverage plot. If this relationship can be formulated, the intercept and the slope for a particular sample size would give an empirical estimate of prediction mean squared error for the samples drawn from the same distributions as the calibration set.

Simulation 3.3. Sample Size and Prediction Uncertainty

The simulation is carried out under the model of Simulation 3.1 Case (a). The simulation parameters are configured as follows: $k = 50$, n_p=400, $\beta_0 = 1$ and $\beta = (\ 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad \cdots \quad 0\ )'$, $\sigma_\epsilon^2 = 0.25$, and $\boldsymbol{\epsilon}_p \neq \mathbf{0}$. The explanatory variables $\dot{\mathbf{X}}_c$ $(n \times k)$ are independent normally distributed $N(0, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma}$ is a diagonal matrix with $\boldsymbol{\sigma}_c^2 = (\ \sigma_{c_1}^2 \quad \sigma_{c_2}^2 \quad \ldots \quad \sigma_{c_k}^2\ )'$ as its diagonal elements. The predictor $\dot{\mathbf{x}}_p$ is simulated from the same distribution as the calibration set.

The number of principal components is set to be $a = 5$. The explanatory variable variances $\boldsymbol{\sigma}_c^2$ are set to fall down exponentially with a big cutoff between the fifth and sixth variables. Define $\boldsymbol{\sigma}_c = \mathbf{c}_1 + \mathbf{c}_2$, where the elements of $\mathbf{c}_1$ are square roots of an exponential function $e^{\mathbf{c}_3}$ sorted in a descending order, and $\mathbf{c}_3$ is a column vector containing k numbers starting from 0 to 3 with an equal step $\frac{3}{k-1}$; $\mathbf{c}_2$ is a column vector whose first five elements equal to 20 and the rest equal to zeros, $\mathbf{c}_2 = (\ 20 \quad 20 \quad 20 \quad 20 \quad 20 \quad 0 \quad \cdots \quad 0\ )'$.

There are twenty sample sizes of the calibration set being studied. They are calculated under the rule that $\frac{1}{n}$ is a series of numbers between 0.02 and 0.005 with an equal step. For each sample size, the simulation has been run repeatedly 5000 times. In each replicate, the principal components regression calculates the squared prediction errors and the leverages for the prediction set, and then an

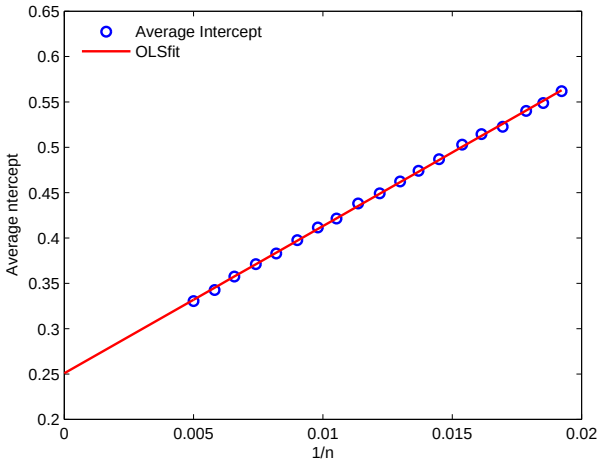
ordinary least squares fit of all the squared prediction errors and the leverages gives a slope and an intercept. In the end, there are 5000 pairs of slopes and intercepts saved for each sample size.

Figure 3.6(a) shows the the relationship between the intercept and $\frac{1}{n}$ in an average squared prediction error against average leverage plot. Figure 3.6(b) gives the relationship between the slope and $\frac{1}{n}$. As expected, the intercept has a linear relationship with $\frac{1}{n}$ in Figure 3.6(a), the red line with an intercept of 0.2526 and a slope of 16.0713 is an ordinary least squares fit of the 20 points. The intercept is consistent with $\sigma_\epsilon^2 = 0.25$ when n is very large. In Figure 3.6(b) the red line with an intercept of -0.1555 and a slope of 178.9176, is also an ordinary least squares fit of all blue points, but the blue line seems to have a slight curvature. The non-zero intercept and the curvature suggest that the slope of the linear relationship between average prediction error and average leverage does not have a simple linear relationship with $\frac{1}{n}$. Hence, the simple empirical estimate would not be easy to obtain.

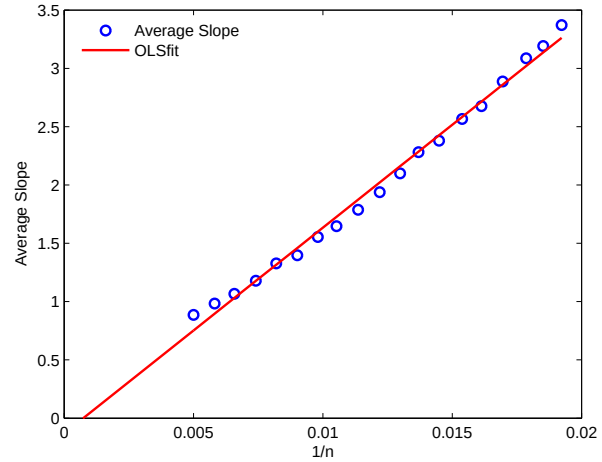
However, combining the two results, we could guess that one part of expected squared bias contributes to the intercept of the linear relationship between squared prediction error and leverage, and the rest goes into the leverage. How is the expected squared bias actually associated with the leverage? As seen from Figure 3.2 average squared prediction error has a linear relationship with average leverage, does this mean the expected squared bias also has a linear relationship with the leverage? We will continue to discuss these questions in next section.

3.3.4 Correlation between Leverage and Bias

In this section, we will investigate the correlation between the leverage and the bias. In principal components regression, the scores for all k explanatory variables can be calculated as $\mathbf{T} = \mathbf{X}_c\mathbf{V}$. $\mathbf{T}$ can be split into two parts $(\mathbf{T}_1 \quad \mathbf{T}_2)$, where $\mathbf{T}_1$ $(n \times a)$ and $\mathbf{T}_2$ $(n \times (k-a))$, and a is the number of principal components. $\mathbf{T}_1$ denotes principal components used in the bilinear model. $\mathbf{T}_2$ is not shown in the bilinear model. It presents the scores of unused components that brings bias. To



(a) Average Intercept vs. $\frac{1}{n}$



(b) Average Slope vs. $\frac{1}{n}$

Figure 3.6: PCR the Relationship between Prediction Mean Squared Error and Sample Size. (a) Average Intercept: the average intercept under a particular sample size n over 5000 replicates against $\frac{1}{n}$. OLSfit: the ordinary least squares fit of average intercept points. (b) Average Slope: the average slope under a particular sample size n over 5000 replicates against $\frac{1}{n}$. OLSfit: the ordinary least squares fit of average slope points.

keep unused components in the model, we shall discover their roles in prediction mean squared error mathematically. The full model can be written as

$$\dot{\mathbf{y}}_c = \alpha + \mathbf{T}_1\boldsymbol{\gamma}_1 + \mathbf{T}_2\boldsymbol{\gamma}_2 + \boldsymbol{\epsilon}_c,$$

where $\mathbf{T}_2$ is independent of $\mathbf{T}_1$, and the loadings $\boldsymbol{\gamma}$ are partitioned into two parts $\boldsymbol{\gamma} = \begin{pmatrix} \boldsymbol{\gamma}_1 \\ \boldsymbol{\gamma}_2 \end{pmatrix}$. $\boldsymbol{\gamma}_1$ is an $a \times 1$ vector, and $\boldsymbol{\gamma}_2$ is a $(k-a) \times 1$ vector. In principal components regression, we choose a principal components, which is equivalent to only keeping $\mathbf{T}_1$ in the model, so the reduced model becomes

$$\dot{\mathbf{y}}_c = \delta + \mathbf{T}_1\boldsymbol{\gamma}_1 + \boldsymbol{\eta}_c,$$

where the unused components has been put into the error term, $\boldsymbol{\eta}_c = \mathbf{T}_2\boldsymbol{\gamma}_2 + \boldsymbol{\epsilon}_c$. Estimated regression coefficients $\hat{\alpha} = \hat{\delta} = \bar{\dot{y}}$ and $\hat{\boldsymbol{\gamma}}_1 = (\mathbf{T}_1'\mathbf{T}_1)^{-1}\mathbf{T}_1'\mathbf{y}_c$. The prediction can be expressed in terms of the full model, $\dot{y}_p = \alpha + \mathbf{t}_{p_1}\boldsymbol{\gamma}_1 + \mathbf{t}_{p_2}\boldsymbol{\gamma}_2 + \epsilon_p$. Since the predicted value is calculated as $\hat{\dot{y}}_p = \hat{\delta} + \mathbf{t}_{p_1}\hat{\boldsymbol{\gamma}}_1$, the difference between the observed value and the predicted value, and the prediction mean squared error can be calculated as

$$\begin{aligned} \dot{y}_p - \hat{\dot{y}}_p &= \alpha + \mathbf{t}_{p_1}\boldsymbol{\gamma}_1 + \mathbf{t}_{p_2}\boldsymbol{\gamma}_2 + \epsilon_p - \hat{\alpha} - \mathbf{t}_{p_1}\hat{\boldsymbol{\gamma}}_1. \\ \mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} &= \sigma_\epsilon^2(1 + \frac{1}{n}) + \mathrm{E}\left[(\mathbf{t}_{p_1}(\boldsymbol{\gamma}_1 - \hat{\boldsymbol{\gamma}}_1) + \mathbf{t}_{p_2}\boldsymbol{\gamma}_2)^2\right]. \end{aligned}$$

In terms of the leverage $h = \mathbf{t}_{p_1}(\mathbf{T}_1'\mathbf{T}_1)^{-1}\mathbf{t}_{p_1}'$, the prediction mean squared error

$$\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \sigma_\epsilon^2\{1 + \frac{1}{n} + \mathbf{t}_{p_1}(\mathbf{T}_1'\mathbf{T}_1)^{-1}\mathbf{t}_{p_1}'\} + (\mathbf{t}_{p_2}\boldsymbol{\gamma}_2)^2.$$

$\mathbf{t}_p = \mathbf{x}_p\mathbf{V} = (\dot{\mathbf{x}}_p - \bar{\dot{\mathbf{x}}})\mathbf{V}$, $\dot{\mathbf{x}}_p \sim N(0, \boldsymbol{\Sigma})$ and $\mathbf{x}_p \sim N(0, (1 + \frac{1}{n})\boldsymbol{\Sigma})$. $\mathbf{t}_p' \sim N(0, (1 + \frac{1}{n})\mathbf{M})$, where

$$\mathbf{M} = \left(\begin{array}{c:c} \underset{a\times a}{\mathbf{M}_{11}} & \underset{(k-a)\times a}{\mathbf{M}_{12}} \\ \hdashline \underset{(k-a)\times a}{\mathbf{M}_{21}} & \underset{(k-a)\times(k-a)}{\mathbf{M}_{22}} \end{array}\right) = \left(\begin{array}{c:c} \mathbf{V}_1'\boldsymbol{\Sigma}\mathbf{V}_1 & \mathbf{V}_1'\boldsymbol{\Sigma}\mathbf{V}_2 \\ \hdashline \mathbf{V}_2'\boldsymbol{\Sigma}\mathbf{V}_1 & \mathbf{V}_2'\boldsymbol{\Sigma}\mathbf{V}_2 \end{array}\right) = \mathbf{V}'\boldsymbol{\Sigma}\mathbf{V}.$$

The predictor score vector can be divided into two parts $\mathbf{t}_p = \begin{pmatrix} \underset{1\times a}{\mathbf{t}_{p_1}} & \underset{1\times(k-a)}{\mathbf{t}_{p_2}} \end{pmatrix}$. $\mathbf{t}_{p_1}$ contains the scores of principal components, and $\mathbf{t}_{p_2}$ contains the scores of unused

predictors. $\mathbf{V}_1$ and $\mathbf{V}_2$ are two parts of eigenvector matrix, $\mathbf{V} = \begin{pmatrix} \mathbf{V}_1 & \mathbf{V}_2 \\ k\times a & k\times(k-a) \end{pmatrix}$. If $\mathbf{V}$ is consist of the eigenvectors of $\mathbf{\Sigma}$, $\mathbf{M}$ would be diagonal, but $\mathbf{V}$ is actually made of the eigenvectors of $\mathbf{X}_c'\mathbf{X}_c$. Since $\mathrm{E}\,(\mathbf{X}_c'\mathbf{X}_c) = (n-1)\mathbf{\Sigma}$ and $\mathbf{V}'\mathbf{X}_c'\mathbf{X}_c\mathbf{V} = \mathbf{\Lambda}$, $\mathbf{V}'\mathbf{\Sigma}\mathbf{V} \approx \frac{1}{n-1}\mathbf{\Lambda}$. The conditional distribution $\mathbf{t}_{p_2}'|\mathbf{t}_{p_1}' \sim N\big(\mathbf{M}_{21}\mathbf{M}_{11}^{-1}\mathbf{t}_{p_1}', (1+\frac{1}{n})(\mathbf{M}_{22} - \mathbf{M}_{21}\mathbf{M}_{11}^{-1}\mathbf{M}_{12})\big)$, which gives $\mathrm{E}\,(\mathbf{t}_{p_2}'\mathbf{t}_{p_2}|\mathbf{t}_{p_1}') = \mathrm{E}\,(\mathbf{t}_{p_2}'|\mathbf{t}_{p_1})'\mathrm{E}\,(\mathbf{t}_{p_2}'|\mathbf{t}_{p_1}') + \mathrm{Var}\,(\mathbf{t}_{p_2}'|\mathbf{t}_{p_1}')$, in detail

$$
\begin{aligned}
&\mathrm{E}\begin{pmatrix} t_{p_{a+1}}^2 & t_{p_{a+1}}t_{p_{a+2}} & \cdots & t_{p_{a+1}}t_{p_k} \\ t_{p_{a+2}}t_{p_{a+1}} & t_{p_{a+2}}^2 & \cdots & t_{p_{a+2}}t_{p_k} \\ \vdots & \vdots & \ddots & \vdots \\ t_{p_k}t_{p_{a+1}} & t_{p_k}t_{p_{a+2}} & \cdots & t_{p_k}^2 \end{pmatrix}|\mathbf{t}_{p_1}' \\
&= (\mathbf{M}_{21}\mathbf{M}_{11}^{-1})\mathbf{t}_{p_1}'\mathbf{t}_{p_1}(\mathbf{M}_{21}\mathbf{M}_{11}^{-1})' + \mathrm{Var}\,(\mathbf{t}_{p_2}'|\mathbf{t}_{p_1}') \\
&= (\mathbf{M}_{21}\mathbf{M}_{11}^{-1})\begin{pmatrix} t_{p_1}^2 & t_{p_1}t_{p_2} & \cdots & t_{p_1}t_{p_a} \\ t_{p_2}t_{p_1} & t_{p_2}^2 & \cdots & t_{p_2}t_{p_a} \\ \vdots & \vdots & \ddots & \vdots \\ t_{p_a}t_{p_1} & t_{p_a}t_{p_2} & \cdots & t_{p_a}^2 \end{pmatrix}(\mathbf{M}_{21}\mathbf{M}_{11}^{-1})' + \mathrm{Var}\,(\mathbf{t}_{p_2}'|\mathbf{t}_{p_1}').
\end{aligned}
$$

Since $\mathbf{\Sigma}$ is symmetric, every diagonal element of $\mathrm{E}\,(\mathbf{t}_{p_2}'\mathbf{t}_{p_2}|\mathbf{t}_{p_1}')$, denoting as $\mathrm{E}\,(t_{p_j}^2|\mathbf{t}_{p_1}')$, for $j = a+1, \cdots, k$, can be written as the sum of the linear functions of squared scores $\sum_{i=1}^{a} c_{ij}t_{p_i}^2$, where c_{ij} is the result of matrix multiplications involved in $\mathbf{M}_{21}$ and $\mathbf{M}_{11}^{-1}$. Hence, $\mathrm{E}\,(\sum_{j=a+1}^{k} t_{p_j}^2|\mathbf{t}_{p_1}')$ can also be written as the sum of the linear functions of squared scores.

$$
\begin{aligned}
\mathrm{E}\,(t_{p_j}^2|\mathbf{t}_{p_1}') &= \sum_{i=1}^{a} c_{ij}t_{p_i}^2 + \text{Constant.} \\
\mathrm{E}\,(\sum_{j=a+1}^{k} t_{p_j}^2|\mathbf{t}_{p_1}') &= \sum_{i=1}^{a} C_i t_{p_i}^2 + \text{Constant.} \qquad (3.15)
\end{aligned}
$$

Every C_i is a linear combination of c_{ij}'s. We are going to use the result (Equation (3.15)) in the simulation analysis.

Simulation 3.4. The Correlation between Leverage and Bias and its Influence in measuring prediction uncertainty

In Simulation 3.4, the regression model is assumed as the same as Simulation 3.3. $k = 50$, $a = 5$, $n = 200$, $n_p = 10000$, $\boldsymbol{\beta} = (\ 1\ \ 1\ \ 1\ \ 1\ \ 1\ \ 0\ \ \cdots\ \ 0\ )$,

$\sigma_\epsilon^2 = 0.25$, and $\boldsymbol{\epsilon}_p = \mathbf{0}$.

$$\boldsymbol{\gamma} = \mathbf{V}'\boldsymbol{\beta} = \begin{pmatrix} v_{11} & v_{21} & \cdots & v_{k1} \\ v_{12} & v_{22} & \cdots & v_{k2} \\ \vdots & \vdots & \ddots & \vdots \\ v_{1k} & v_{2k} & \cdots & v_{kk} \end{pmatrix} = \begin{pmatrix} 1 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \left(\begin{array}{c} \sum_{i=1}^{a} v_{i1} \\ \vdots \\ \sum_{i=1}^{a} v_{ia} \\ \hdashline \sum_{i=1}^{a} v_{i(a+1)} \\ \vdots \\ \sum_{i=1}^{a} v_{ik} \end{array}\right) = \begin{pmatrix} \boldsymbol{\gamma}_1 \\ \boldsymbol{\gamma}_2 \end{pmatrix}.$$

We explore two different cases of explanatory variable variance. Figure 3.7 gives plots of squared bias against leverage. Figure 3.7(a) shows the case where $\boldsymbol{\sigma}_c$ is a vector containing 50 elements starting from 10 to 0.1 decreasing with an equal step. In Figure 3.7(b), the first five elements of $\boldsymbol{\sigma}_c$ are set to be 10, and the rest are all equal to 1. $\mathbf{t}_{p_2}\boldsymbol{\gamma}_2$ has the variance $\frac{1}{n-1}\sum_{i=a+1}^{k}\lambda_i^2\gamma_i^2$. The leverage

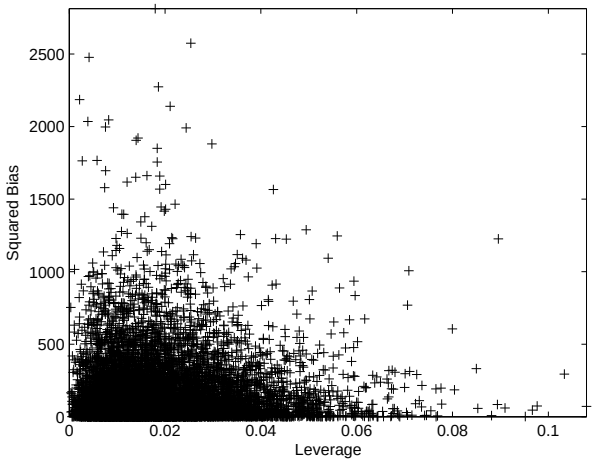
$$h = \mathbf{t}_{p_1}(\mathbf{T}_1'\mathbf{T}_1)^{-1}\mathbf{t}_{p_1}' = \mathbf{t}_{p_1}\boldsymbol{\Lambda}_1^{-1}\mathbf{t}_{p_1}' = \sum_{i=1}^{a}\frac{t_{p_i}^2}{\lambda_i}. \tag{3.16}$$

$$\begin{aligned} \text{bias}^2 &= (\mathbf{t}_{p_2}\boldsymbol{\gamma}_2)^2 = (\sum_{j=a+1}^{k} t_{p_j}\sum_{i=1}^{a} v_{ij})^2 \\ &= (\sum_{j=a+1}^{k} t_{p_j}^2 + 2\sum_{j=a+1}^{k}\sum_{\substack{j\neq r \\ j=a+1}}^{k} t_{p_j}t_{p_r})(\sum_{i=1}^{a} v_{ij}^2 + 2\sum_{i=1}^{a}\sum_{\substack{j\neq l \\ j=a+1}}^{k} v_{ij}v_{il}). \end{aligned} \tag{3.17}$$

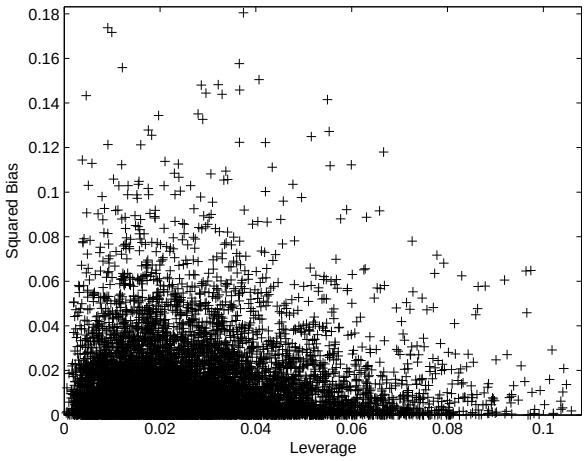
Therefore, using Equation (3.15), and putting all other terms into a constant term,

$$\mathrm{E}\,(\text{bias}^2) = \sum_{i=1}^{a} C_i t_{p_i}^2 + \text{Constant}, \tag{3.18}$$

which suggests the expected squared bias approximate to a linear function of the leverage. When eigenvalues are close or equal, the squared bias and the leverage are linearly correlated. If eigenvalues are quite different, the correlation is not linear. $h = \sum_{i=1}^{a}\frac{t_{p_i}^2}{\lambda_i}$ is an ellipsoid. $\text{bias}^2 = \sum_{i=1}^{a} C_i t_{p_i}^2$ is also an ellipsoid, expanding the leverage ellipsoid at the rate of $\lambda_i C_i$ in the direction of t_{p_i}. Figure 3.7(a) shows the non-linear association between squared bias and leverage. Figure 3.7(b) is an example of the ideal situation when squared bias is a linear combination of the leverage elements. The different y-axis scales suggests that the squared bias in



(a) explanatory variables with stepwise decreasing variances.



(b) the variances of first five explanatory variables equal to 10, and the others are equal to 1.

Figure 3.7: PCR the Relationship between the Squared Bias and the Leverage.

Figure 3.7(b) is less noisy than that of Figure 3.7(a), because the linear function is the simplest relationship between expected squared bias and leverage.

Taking a two-variable case as a mathematical example to show the simplest relationship, where $k = 2$, $\boldsymbol{\beta} = (\ 1 \quad 0\ )'$, only one variable is chosen as the principal component, and $\boldsymbol{\sigma}_c = (\ \sigma_{c_1}^2 \quad \sigma_{c_2}^2\ )$.

$$\begin{aligned}
\mathbf{V} &= \begin{pmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{pmatrix} \qquad \text{and} \qquad \boldsymbol{\Sigma} = \begin{pmatrix} \sigma_{c_1}^2 & 0 \\ 0 & \sigma_{c_2}^2 \end{pmatrix}. \\
\mathbf{V}'\boldsymbol{\Sigma}\mathbf{V} &= \begin{pmatrix} v_{11}^2\sigma_{c_1}^2 + v_{21}^2\sigma_{c_2}^2 & v_{11}v_{12}\sigma_{c_1}^2 + v_{21}v_{22}\sigma_{c_2}^2 \\ v_{11}v_{12}\sigma_{c_1}^2 + v_{21}v_{22}\sigma_{c_2}^2 & v_{12}^2\sigma_{c_1}^2 + v_{22}^2\sigma_{c_2}^2 \end{pmatrix}. \\
\mathrm{E}\,(t_{p_2}|t_{p_1}) &= \Big(\frac{v_{11}v_{12}\sigma_{c_1}^2 + v_{21}v_{22}\sigma_{c_2}^2}{v_{11}^2\sigma_{c_1}^2 + v_{21}^2\sigma_{c_2}^2}\Big)t_{p_1}. \\
\mathrm{Var}\,(t_{p_2}|t_{p_1}) &= \frac{\sigma_{c_1}^2\sigma_{c_2}^2(v_{12}v_{21} - v_{11}v_{22})^2}{v_{11}^2\sigma_{c_1}^2 + v_{21}^2\sigma_{c_2}^2}. \\
\mathrm{E}\,(t_{p_2}^2|t_{p_1}) &= \{\mathrm{E}\,(t_{p_2}|t_{p_1})\}^2 + \mathrm{Var}\,(t_{p_2}|t_{p_1}) \\
&= \Big(\frac{v_{11}v_{12}\sigma_{c_1}^2 + v_{21}v_{22}\sigma_{c_2}^2}{v_{11}^2\sigma_{c_1}^2 + v_{21}^2\sigma_{c_2}^2}\Big)^2 t_{p_1}^2 + \frac{\sigma_{c_1}^2\sigma_{c_2}^2(v_{12}v_{21} - v_{11}v_{22})^2}{v_{11}^2\sigma_{c_1}^2 + v_{21}^2\sigma_{c_2}^2}.
\end{aligned}$$

Since $\boldsymbol{\gamma} = \begin{pmatrix} v_{11} \\ v_{12} \end{pmatrix}$ and $\text{bias}^2 = v_{12}^2 t_{p_2}^2$, and the leverage $h = \frac{t_{p_1}^2}{\lambda_1^2}$ where λ_1 is the biggest eigenvalue of $\mathbf{X}_c'\mathbf{X}_c$, the expected squared bias can be written as a linear function of the leverage

$$\mathrm{E}\,(\text{bias}^2) = v_{12}^2\{\Big(\frac{v_{11}v_{12}\sigma_{c_1}^2 + v_{21}v_{22}\sigma_{c_2}^2}{v_{11}^2\sigma_{c_1}^2 + v_{21}^2\sigma_{c_2}^2}\Big)^2\lambda_1^2 h + \frac{\sigma_{c_1}^2\sigma_{c_2}^2(v_{12}v_{21} - v_{11}v_{22})^2}{v_{11}^2\sigma_{c_1}^2 + v_{21}^2\sigma_{c_2}^2}\}. \quad (3.19)$$

Equation (3.19) suggests the expected squared bias has a positive relationship with the leverage, so the slope of squared prediction error against leverage is always steeper than that of the ordinary least squares type expression as shown in Figure 3.4 and Figure 3.5.

In simple cases where the eigenvalues of the sample covariance matrix are close or equal, the expected squared bias has a linear relationship with the leverage. However, in all other cases, the relationship between expected bias and leverage is difficult to formulate mathematically, and in more complicated circumstances, principal components chosen in the regression process may not be the principal

components assumed in the true model. Therefore, it hardly has a chance to establish the relationship between prediction uncertainty and leverage via the correlation between expected squared bias and leverage.

3.4 Summary

The term 'bias' discussed does not mean or include systematic errors in the measurements of explanatory and response variables, which may be caused by measurement instrument, environment change, raw data treatment, and so on. It is only referred to the bias in the statistical modelling process. In this chapter we verify that the expected squared bias is missing from the ordinary least squares type prediction mean squared error. The ordinary least squares type prediction mean squared error with the estimated regression error variance from the tuning set seems a reasonable quantification of principal components prediction uncertainty because it compensates the omission of unused explanatory variables to some extend. To explore other possible ways to model prediction uncertainty, we try to use sample size to find an empirical estimate for the relationship between squared prediction error and leverage, but it fails. We also investigate the correlation between squared bias and leverage, and its influence upon squared prediction error, hoping to model the squared bias based on the leverage. However, we find, even in the simplest case where the squared bias is assumed to be linear with the leverage, the linear relationship cannot be seen ideally from the simulation result since the closed mathematical form of the relationship between expected squared bias and leverage is unclear, although the expected squared bias has a positive relationship with the leverage. Therefore, it would be suggested to use the ordinary least squares type prediction mean squared error with the estimated regression error variance from the tuning set as the measurement of principal components prediction uncertainty, as an alternative to the empirical estimates RMSEP and RMSECV.

Chapter 4

Partial Least Squares Regression Prediction Uncertainty

This chapter is a foundation of partial least squares regression prediction uncertainty study. It introduces the basic theory of univariate partial least squares regression, and presents univariate partial least squares algorithms being used in the thesis. It unites the mathematical forms and notations, and clarifies the whole area, which is one of the main contributions of the thesis.

- Section 4.1 introduces orthogonal scores and orthogonal loadings univariate partial least squares algorithms.
- Existing theoretical works for partial least squares regression prediction uncertainty are summarised in Section 4.2, including root mean squared error of prediction, ordinary least squares type expression, linearisation based methods, re-sampling methods, and U-deviation methods.

4.1 Partial Least Squares Regression Algorithms

The origin of partial least squares modelling can date back to the 1970s. Wold (1966) calculates principal components using an iterative process, and Wold (1973) gives the non-linear iterative partial least squares algorithm (NIPALS), calculating

principal components with an iterative sequence of simple regressions using ordinary least squares regression. The main difference between principal components regression and partial least squares regression is whether the response variable participates in the construction of factors. Principal components regression chooses explanatory variables with large variance as principal components, although the selection of the number of principal components actually involves the response variable via the cross-validation. Partial least squares regression constructs the factors using the maximisation of the covariance between explanatory and response variables as the criterion. There are many partial least squares algorithms developed. Andersson (2009) studies the performance of nine univariate partial least squares algorithms. To keep it simple and easy to read, only algorithms relevant to the thesis are introduced.

4.1.1 Orthogonal Scores Algorithms

Algorithm 4.1. Non-linear Iterative Partial Least Squares Algorithm

Partial Least Squares Regression compresses the dimensions of both centred explanatory variables $\mathbf{X}_c$ and centred response variables $\mathbf{Y}_c$ separately, where $\mathbf{Y}_c$ is a multivariate variable. It constructs new variables by taking linear combinations of original variables. It has been designed that both $\mathbf{X}_c$ and $\mathbf{Y}_c$ can be related to the scores $\mathbf{T}$. $\mathbf{W}$ is a matrix of loading weights. Let a row vector $\bar{\dot{\mathbf{x}}}$ denote the mean of $\dot{\mathbf{X}}_c$, and let a row vector $\bar{\dot{\mathbf{y}}}$ denote the mean of $\dot{\mathbf{Y}}_c$. The centred explanatory variables and the centred response variable can be written as $\mathbf{X}_c = \dot{\mathbf{X}}_c - \mathbf{1}\bar{\dot{\mathbf{x}}}$ and $\mathbf{Y}_c = \dot{\mathbf{Y}}_c - \mathbf{1}\bar{\dot{\mathbf{y}}}$. Höskuldsson (1988) analyses the non-linear iterative partial least squares algorithm. Set $\mathbf{u}_1$ to be the first column of $\mathbf{Y}_c$. Let $\dot{\mathbf{X}}_{c_0}$ and $\mathbf{Y}_{c_0}$ denote the original centered data $\mathbf{X}_c$ and $\mathbf{Y}_c$. The number of factors is set to be a, and $i = 1, \ldots, a$.

- $\mathbf{w}_i = \mathbf{X}'_{c_{i-1}}\mathbf{u}_i/(\mathbf{u}'_i\mathbf{u}_i)$ and scale $\mathbf{w}_i$ to be a unit vector, $\mathbf{w}_i = \mathbf{w}_i/||\mathbf{w}_i||$.
- $\mathbf{t}_i = \mathbf{X}_{c_{i-1}}\mathbf{w}_i$.
- $\hat{\mathbf{c}}_i = \mathbf{Y}'_{c_{i-1}}\mathbf{t}_i/(\mathbf{t}'_i\mathbf{t}_i)$.

- $\mathbf{u}_i = \mathbf{Y}'_{c_{i-1}}\hat{\mathbf{c}}_i/(\hat{\mathbf{c}}'_i\hat{\mathbf{c}}_i)$ and scale $\hat{\mathbf{c}}_i$ to be of length one $\hat{\mathbf{c}}_i = \hat{\mathbf{c}}_i/||\hat{\mathbf{c}}_i||$.
- If $\frac{||\mathbf{u}_{i-1}-\mathbf{u}_i||}{||\mathbf{u}_i||} < \kappa$ (with κ for example set to 10^{-6} or 10^{-8}) the convergence is achieved, then go to next step else the first step.
- X-loadings: $\hat{\mathbf{p}}_i = \mathbf{X}'_{c_{i-1}}\mathbf{t}_i/(\mathbf{t}'_i\mathbf{t}_i)$.
- Y-loadings: $\hat{\mathbf{q}}_i = \mathbf{Y}'_{c_{i-1}}\mathbf{u}_i/(\mathbf{u}'_i\mathbf{u}_i)$.
- Regression ($\mathbf{u}_i$ upon $\mathbf{t}_i$): $d_i = \mathbf{u}'_i\mathbf{t}_i/(\mathbf{t}'_i\mathbf{t}_i)$.
- Residual matrices: $\mathbf{X}_{c_i} = \mathbf{X}_{c_{i-1}} - \mathbf{t}_i\hat{\mathbf{p}}'_i$ and $\mathbf{Y}_{c_i} = \mathbf{Y}_{c_{i-1}} - d_i\mathbf{t}_i\hat{\mathbf{c}}'_i$

The next iteration begins with the residual matrices $\mathbf{X}_{c_i}$ and $\mathbf{Y}_{c_i}$ calculated from the previous iteration. The iterations stop when the residual matrix $\mathbf{X}_{c_i}$ becomes a zero matrix. To study the situation when the algorithm reaches convergence, let us have a look at $\mathbf{u}_i$, $\hat{\mathbf{c}}_i$, $\mathbf{t}_i$ and $\mathbf{w}_i$.

$$\begin{aligned}
\mathbf{u}_i &= \mathbf{Y}_{c_{i-1}}\hat{\mathbf{c}}_i/(\hat{\mathbf{c}}'_i\hat{\mathbf{c}}_i) \\
&= \mathbf{Y}_{c_{i-1}}\mathbf{Y}'_{c_{i-1}}\mathbf{t}_i/\{(\hat{\mathbf{c}}'_i\hat{\mathbf{c}}_i)(\mathbf{t}'_i\mathbf{t}_i)\} \\
&= \mathbf{Y}_{c_{i-1}}\mathbf{Y}'_{c_{i-1}}\mathbf{X}_{c_{i-1}}\mathbf{w}_i/\{(\hat{\mathbf{c}}'_i\hat{\mathbf{c}}_i)(\mathbf{t}'_i\mathbf{t}_i)(\mathbf{w}'_i\mathbf{w}_i)\} \\
&= \mathbf{Y}_{c_{i-1}}\mathbf{Y}'_{c_{i-1}}\mathbf{X}_{c_{i-1}}\mathbf{X}'_{c_{i-1}}\mathbf{u}_{i-1}/\{(\hat{\mathbf{c}}'_i\hat{\mathbf{c}}_i)(\mathbf{t}'_i\mathbf{t}_i)(\mathbf{w}'_i\mathbf{w}_i)(\mathbf{u}'_{i-1}\mathbf{u}_{i-1})\}.
\end{aligned}$$

Similarly, we have

$$\begin{aligned}
\hat{\mathbf{c}}_i &= \mathbf{Y}'_{c_{i-1}}\mathbf{X}_{c_{i-1}}\mathbf{X}_{c_{i-1}}\mathbf{Y}_{c_i}\hat{\mathbf{c}}_{i-1}/\{(\mathbf{t}'_i\mathbf{t}_i)(\mathbf{w}'_i\mathbf{w}_i)(\mathbf{u}'_{i-1}\mathbf{u}_{i-1})(\hat{\mathbf{c}}'_{i-1}\hat{\mathbf{c}}_{i-1})\}, \\
\mathbf{t}_i &= \mathbf{X}_{c_{i-1}}\mathbf{X}'_{c_{i-1}}\mathbf{Y}_{c_{i-1}}\mathbf{Y}'_{c_{i-1}}\mathbf{t}_{i-1}/\{(\mathbf{w}'_i\mathbf{w}_i)(\mathbf{u}'_{i-1}\mathbf{u}_{i-1})(\hat{\mathbf{c}}'_{i-1}\hat{\mathbf{c}}_{i-1})(\mathbf{t}'_{i-1}\mathbf{t}_{i-1})\}, \\
\mathbf{w}_i &= \mathbf{X}'_{c_{i-1}}\mathbf{Y}_{c_{i-1}}\mathbf{Y}'_{c_{i-1}}\mathbf{X}'_{c_{i-1}}\mathbf{w}_{i-1}/\{(\mathbf{u}'_{i-1}\mathbf{u}_{i-1})(\hat{\mathbf{c}}'_{i-1}\hat{\mathbf{c}}_{i-1})(\mathbf{t}'_{i-1}\mathbf{t}_{i-1})(\mathbf{w}'_{i-1}\mathbf{w}_{i-1})\}.
\end{aligned}$$

At the i-th iteration when the algorithm converges we can write

$$\begin{aligned}
\mathbf{Y}_{c_{i-1}}\mathbf{Y}'_{c_{i-1}}\mathbf{X}_{c_{i-1}}\mathbf{X}'_{c_{i-1}}\mathbf{u}_i &= \lambda_u\mathbf{u}_i, \\
\mathbf{Y}'_{c_{i-1}}\mathbf{X}_{c_{i-1}}\mathbf{X}'_{c_{i-1}}\mathbf{Y}_{c_{i-1}}\hat{\mathbf{c}}_i &= \lambda_c\hat{\mathbf{c}}_i, \\
\mathbf{X}_{c_{i-1}}\mathbf{X}'_{c_{i-1}}\mathbf{Y}_{c_{i-1}}\mathbf{Y}'_{c_{i-1}}\mathbf{t}_i &= \lambda_t\mathbf{t}_i, \\
\mathbf{X}'_{c_{i-1}}\mathbf{Y}_{c_{i-1}}\mathbf{Y}'_{c_{i-1}}\mathbf{X}_{c_{i-1}}\mathbf{w}_i &= \lambda_w\mathbf{w}_i.
\end{aligned}$$

The power method shows that λ_u, λ_c, λ_t and λ_w are the maximum eigenvalues of the eigenvalue problem. The vectors $\mathbf{u}$, $\hat{\mathbf{c}}$, $\mathbf{t}$ and $\mathbf{w}$ are the eigenvectors corresponding to the maximum eigenvalues. The algorithm computes the maximum eigenvalues and associated eigenvectors of the matrices

$$\begin{array}{ll} \mathbf{Y}_{c_{i-1}}\mathbf{Y}'_{c_{i-1}}\mathbf{X}_{c_{i-1}}\mathbf{X}'_{c_{i-1}} & \mathbf{Y}'_{c_{i-1}}\mathbf{X}_{c_{i-1}}\mathbf{X}'_{c_{i-1}}\mathbf{Y}_{c_{i-1}} \\ \mathbf{X}_{c_{i-1}}\mathbf{X}'_{c_{i-1}}\mathbf{Y}_{c_{i-1}}\mathbf{Y}'_{c_{i-1}} & \mathbf{X}'_{c_{i-1}}\mathbf{Y}_{c_{i-1}}\mathbf{Y}'_{c_{i-1}}\mathbf{X}_{c_{i-1}}. \end{array}$$

The eigenvectors of i-th iteration are used to calculate the new residual matrices $\mathbf{X}_{c_i}$ and $\mathbf{Y}_{c_i}$. In the computations only the eigenvector of $\mathbf{X}'_{c_{i-1}}\mathbf{Y}_{c_{i-1}}\mathbf{Y}'_{c_{i-1}}\mathbf{X}_{c_{i-1}}$ is needed, and the others may be computed as in the algorithm. The maximisation criterion of NIPALS is given by

$$\begin{array}{rc} \text{maximise} & \text{Cov}\left(\mathbf{X}_{c_{i-1}}\mathbf{w}_i, \mathbf{Y}_{c_{i-1}}\hat{\mathbf{q}}_i\right) \\ \text{subject to} & \mathbf{w}'_i\mathbf{w}_i = \hat{\mathbf{q}}'_i\hat{\mathbf{q}}_i = 1 \\ & \text{Cov}\left(\mathbf{X}_{c_{i-1}}\mathbf{w}_i, \mathbf{X}_{c_{j-1}}\mathbf{w}_j\right) = 0, \text{ for } i \neq j \end{array} .$$

The vectors $\mathbf{w}_i$ and $\mathbf{c}_i$ in the algorithm satisfy the maximisation. The vectors $\mathbf{u}_i$ and $\mathbf{t}_i$ have the property that

$$\begin{aligned} \{\text{Cov}\left(\mathbf{t}_i, \mathbf{u}_i\right)\}^2 &= \{\text{Cov}\left(\mathbf{X}_{c_{i-1}}\mathbf{w}_i, \mathbf{Y}_{c_{i-1}}\hat{\mathbf{c}}_i\right)\}^2 \\ &= \{\mathbf{w}'_i\text{Cov}\left(\mathbf{X}_{c_{i-1}}, \mathbf{Y}_{c_{i-1}}\right)\hat{\mathbf{c}}_i\}^2 \\ &= (\mathbf{w}'_i\mathbf{X}'_{c_{i-1}}\mathbf{Y}_{c_{i-1}}\hat{\mathbf{c}}_i)^2. \end{aligned}$$

The bilinear PLS models with centred data can be written as

$$\begin{aligned} \mathbf{T} &= \mathbf{X}_c\mathbf{W}, \\ \mathbf{X}_c &= \mathbf{T}\mathbf{P}' + \mathbf{E} = \sum_{i=1}^{a}\mathbf{t}_i\mathbf{p}'_i + \mathbf{E}, \\ \mathbf{Y}_c &= \mathbf{U}\mathbf{C}' + \mathbf{F}^* = \sum_{i=1}^{a}\mathbf{u}_i\mathbf{c}'_i + \mathbf{F}^*. \end{aligned}$$

Rosipal and Krämer (2006) points out that if we assume the x-scores $\{\mathbf{t}_i, i = 1, \cdots, k\}$ are good predictors of $\mathbf{Y}_c$, and there exists a linear inner relation between $\mathbf{t}$ and $\mathbf{u}$, that is,

$$\mathbf{U} = \mathbf{T}\mathbf{D} + \mathbf{L},$$

where $\mathbf{D}$ is a $p \times p$ diagonal matrix and $\mathbf{L}$ denotes the residual matrix, then

$$\mathbf{Y}_c = \mathbf{TDC}' + (\mathbf{LC}' + \mathbf{F}^*).$$

Define $\mathbf{Q} = \mathbf{CD}'$ and the new residuals matrix $\mathbf{F}$. We have

$$\mathbf{Y}_c = \mathbf{TQ}' + \mathbf{F}.$$

The linear latent relationship between scores $\mathbf{t}$ and $\mathbf{u}$ build a bridge from the response variable to x-scores $\mathbf{T}$ directly, which suggests an alternative form of NIPALS algorithm (See Algorithm 4.2).

To associate the score $\mathbf{T}$ with the original matrix $\mathbf{X}_c$, De Jong (1993) assumes an alternative weight matrix $\mathbf{R}$,

$$\mathbf{T} = \mathbf{X}_c\mathbf{R} \qquad\qquad \mathbf{t}_i = \mathbf{X}_c\mathbf{r}_i, \quad i = 1, \cdots, a.$$

$\mathbf{R}$ can be computed from the regression of $\mathbf{T}$ on $\mathbf{X}_c$

$$\begin{aligned} \mathbf{R} = \mathbf{X}_c^+\mathbf{T} &= (\mathbf{X}_c'\mathbf{X}_c)^-\mathbf{X}_c'\mathbf{T} = (\mathbf{X}_c'\mathbf{X}_c)^-(\mathbf{TP}')'(\mathbf{T}')^{-1} \\ &= (\mathbf{X}_c'\mathbf{X}_c)^-\mathbf{P}(\mathbf{T}'\mathbf{T})^{-1}, \end{aligned}$$

where the superscript $^-$ indicates the Moore-Penrose pseudo-inverse and $^+$ indicates any generalised inverse. $\mathbf{P}'\mathbf{R} = \mathbf{I}$, where $\mathbf{I}$ $(a \times a)$ is the identity matrix. Another expression is given by Helland (1988), $\mathbf{R} = \mathbf{W}(\mathbf{P}'\mathbf{W})^{-1}$, which follows the observation that $\mathbf{R}$ and $\mathbf{W}$ share the same column space. Hence estimated regression coefficients $\hat{\boldsymbol{\beta}} = \mathbf{R}\hat{\mathbf{Q}}' = \mathbf{W}(\hat{\mathbf{P}}'\mathbf{W})^{-1}\hat{\mathbf{Q}}'$. The regression model of $\mathbf{Y}_c$ on $\mathbf{X}_c$ directly can be written as

$$\begin{aligned} \mathbf{Y}_c &= \mathbf{TQ}' + \mathbf{F} = \mathbf{X}_c\mathbf{RQ}' + \mathbf{E} \\ &= \mathbf{X}_c\boldsymbol{\beta} + \mathbf{E}. \end{aligned}$$

Algorithm 4.2. Orthogonal Scores Algorithm

The orthogonal scores algorithm by Martens and Næs (1991) is widely used as a stable and simple algorithm. Transformed from the NIPALS algorithm, it uses the latent linear relationship to connect the explanatory and response variables

directly, instead of considering explanatory and response variable scores separately. When the number of factors is chosen to be a, the i-th iteration of the algorithm gives the results of the i-th factor, where $i = 1, \cdots, a$. Höskuldsson (1988) shows the deflation of the response variable in univariate NIPALS is not necessary. For simplicity the univariate orthogonal scores algorithm does not include the deflation of the response variable.

1. **Calibration**

   - $\mathbf{w}_i = \mathbf{X}'_{c_i}\mathbf{y}_c$.
   - $\mathbf{t}_i = \mathbf{X}_{c_i}\mathbf{w}_i$.
   - $\hat{\mathbf{p}}_i = \mathbf{X}'_{c_i}\mathbf{t}_i/(\mathbf{t}'_i\mathbf{t}_i)$.
   - $\hat{q}_i = \mathbf{y}'_c\mathbf{t}_i/(\mathbf{t}'_i\mathbf{t}_i)$.
   - $\mathbf{X}_{c_{i+1}} = \mathbf{E}_i = \mathbf{X}_{c_i} - \mathbf{t}_i\hat{\mathbf{p}}'_i$.

   The $(i+1)$-th calibration matrix is defined as the i-th residual matrix estimate. The algorithm is set to start from the centred data that gives $\mathbf{X}_{c_1} = \mathbf{X}_c$. The column vector $\mathbf{w}_i$ $(k \times 1)$ is the weight vector defined by the covariance between $\mathbf{X}_{c_i}$ and $\mathbf{y}_c$. The score matrix $\mathbf{T} = (\ \mathbf{t}_1 \quad \mathbf{t}_2 \quad \cdots \quad \mathbf{t}_a\ )$ is orthogonal. The explanatory variables and the response variable are connected by latent variable $\mathbf{t}_i$ $(n \times 1)$ with loadings $\hat{\mathbf{p}}_i$ $(k \times 1)$ and $\hat{q}_i$. $\mathbf{E}_i$ $(n \times k)$ is the x-residual matrix. The weight matrix $\mathbf{W} = (\ \mathbf{w}_1 \quad \mathbf{w}_2 \quad \cdots \quad \mathbf{w}_a\ )$, and the x-loading matrix $\hat{\mathbf{P}} = (\ \hat{\mathbf{p}}_1 \quad \hat{\mathbf{p}}_2 \quad \cdots \quad \hat{\mathbf{p}}_a\ )$. In the first step, if $\mathbf{w}_i$ is scaled to be of length one $\mathbf{w}_i = \mathbf{w}_i/(\mathbf{w}'_i\mathbf{w}_i)$, the algorithm would become more stable, and it would be easier to compare scores, but the normalisation will not change regression coefficient estimate

$$\hat{\boldsymbol{\beta}} = \mathbf{W}(\hat{\mathbf{P}}'\mathbf{W})^{-1}\hat{\mathbf{q}}. \tag{4.1}$$

   Hence, the scores can also be written as $\mathbf{T} = \mathbf{X}_c\mathbf{W}(\hat{\mathbf{P}}'\mathbf{W})^{-1}$. Romera (2010) uses the non-normalised orthogonal scores algorithm to develop a new local linearisation method in the study of partial least squares regression prediction

uncertainty (See Section 5.1). The y-loadings $\hat{\mathbf{q}}$ is defined as a $a \times 1$ column vector for mathematical convenience, thus the bilinear model for univariate partial least squares regression is written as

$$\begin{aligned} \mathbf{X}_c &= \mathbf{T}\mathbf{P}' + \mathbf{E} \\ \mathbf{Y}_c &= \mathbf{T}\mathbf{q} + \mathbf{F} \end{aligned}$$

2. **Prediction**

   A prediction $\hat{\dot{y}}_p$ can be produced via the score of $\mathbf{x}_p$ $(1 \times k)$. Different from the calibration where $\mathbf{t}_i$ is assumed to be a column vector, the predictor score $\mathbf{t}_p$ is set to be a row vector in order to bring mathematical convenience, $\mathbf{t}_p = (\ t_{p_1} \quad t_{p_2} \quad \cdots \quad t_{p_a}\ )$.

   - $t_{p_i} = \mathbf{x}_{p_i}\mathbf{w}_i$,
   - $\mathbf{x}_{p_{i+1}} = \mathbf{x}_{p_i} - t_{p_i}\hat{\mathbf{p}}_i'$,

   where $\mathbf{x}_{p_1} = \dot{\mathbf{x}}_p - \bar{\dot{\mathbf{x}}}$, so $\hat{\dot{y}}_p = \bar{\dot{y}} + \mathbf{t}_p\hat{\mathbf{q}}'$. Equivalently, $\mathbf{t}_p = \mathbf{x}_p\mathbf{W}(\hat{\mathbf{P}}'\mathbf{W})^{-1}$.

4.1.2 Orthogonal Loadings Algorithms

Algorithm 4.3. Orthogonal Loadings Algorithm

Martens and Næs (1991) also gives an orthogonal loadings algorithm. Similarly to principal components regression, the weight matrix $\mathbf{W}$ is assumed to be equal to the x-loadings $\hat{\mathbf{P}}$. For a single response variable, the deflation of $\mathbf{y}_c$ is unnecessary. The univariate orthogonal loadings algorithm carries on.

1. **Calibration**

   - $\hat{\mathbf{p}}_i = \mathbf{X}_{c_i}'\mathbf{y}_c$.
   - $\mathbf{t}_i = \mathbf{X}_{c_i}\hat{\mathbf{p}}_i$.
   - $\mathbf{T}_i = (\ \mathbf{t}_i \quad \cdots \quad \mathbf{t}_i\ )$.
   - $\hat{\mathbf{q}} = (\mathbf{T}_i'\mathbf{T}_i)^{-1}\mathbf{T}_i'\mathbf{y}_c$.

- $\mathbf{X}_{c_{i+1}} = \mathbf{E}_i = \mathbf{X}_{c_i} - \mathbf{t}_i\hat{\mathbf{p}}_i'$.

Usually, $\hat{\mathbf{p}}_i$ $(k\times 1)$ in the first step is scaled to be of length one $\hat{\mathbf{p}}_i = \hat{\mathbf{p}}_i/(\hat{\mathbf{p}}_i'\hat{\mathbf{p}}_i)$. To be consistent with Algorithm 4.2, the normalisation is not written in the procedure because practitioners can choose to use it or not. The loading matrix $\hat{\mathbf{P}}$ is orthogonal.

2. **Prediction**

- $t_{p_i} = \mathbf{x}_{p_i}\hat{\mathbf{p}}_i$,
- $\mathbf{x}_{p_{i+1}} = \mathbf{x}_{p_i} - t_{p_i}\hat{\mathbf{p}}_i'$,

where $\mathbf{x}_{p_1} = \dot{\mathbf{x}}_p - \bar{\dot{\mathbf{x}}}$, so $\hat{y}_p = \bar{\dot{y}} + \mathbf{t}_p\hat{\mathbf{q}}$. Equivalently, $\hat{\boldsymbol{\beta}} = \mathbf{W}\hat{\mathbf{q}}$ and $\mathbf{t}_p = \mathbf{x}_p\mathbf{W}$.

Helland (1988) has proved the orthogonal loadings algorithm is equivalent to the orthogonal scores algorithm. Helland (1988) also gives another form of the orthogonal loadings algorithm, which is employed by Denham (1997) to study the linearisation methods of prediction uncertainty (See Section 4.2.3).

Algorithm 4.4. Univariate Orthogonal Loadings Algorithm Proposed in Helland (1988)

$\mathbf{S}_{xx}$ and $\mathbf{s}_{xy}$ are defined as the sums of product matrices, $\mathbf{S}_{xx} = \mathbf{X}_c'\mathbf{X}_c$ and $\mathbf{s}_{xy} = \mathbf{X}_c'\mathbf{y}_c$. Define $\mathbf{H}_0 = \mathbf{0}$ $(k \times k)$. For $i = 1, \ldots, a$,

1. **Calibration**

- $\mathbf{w}_i = (\mathbf{I} - \mathbf{S}_{xx}\mathbf{H}_{i-1})\mathbf{s}_{xy}$,
- $\tilde{\mathbf{w}}_i = (\mathbf{I} - \mathbf{H}_{i-1}\mathbf{S}_{xx})\mathbf{w}_i$,
- $\mathbf{H}_i = \mathbf{H}_{i-1} + \tilde{\mathbf{w}}_i\tilde{\mathbf{w}}_i'/(\tilde{\mathbf{w}}_i'\mathbf{S}_{xx}\tilde{\mathbf{w}}_i)$,
- $\hat{\boldsymbol{\beta}}_i = \mathbf{H}_i\mathbf{s}_{xy}$,

2. **Prediction**

- $\hat{y}_p = \bar{\dot{y}} + \mathbf{x}_p\hat{\boldsymbol{\beta}}$

4.2 Partial Least Squares Regression Prediction Uncertainty Literature Review

The quantification methods for partial least squares prediction uncertainty suggested in the literature are quite varied, can lead to quite different answers, and often involve doubtful approximations. Given the confusion, it is not surprising that much of the otherwise very good software that is available for implementing multivariate calibration is deficient when it comes to prediction uncertainty. The literature view clarifies these methods used in the literature, and gives clear guidance in exploring the quantification of prediction uncertainty in partial least squares regression. The methods discussed in the thesis focus on prediction mean squared error using a frequentist approach, as does the majority of the existing literature. Another possible approach that could be taken would be to work in the Bayesian framework, where log predictive score could be studied. This considers the fit of the whole distribution, and is less sensitive to outliers than prediction mean squared error. The Bayesian approach will not be studied here. It would have advantages, but at the price of additional complexity.

4.2.1 Simple Empirical Estimates: RMSEP and RMSECV

The root mean squared error of prediction (RMSEP) is a simple empirical estimate of prediction uncertainty (see Section 1.1). A potential weakness of RMSEP is that the same prediction standard error is attached to all predictions. Olivieri et al. (2006) summarises works of using RMSEP in partial least squares regression. Similarly to principal components regression, the root mean square error of cross-validation (RMSECV) is also a standard empirical estimate of prediction uncertainty (See Section 3.2.1). The root mean square error of cross-validation is defined in Equation 3.7

$$\text{RMSECV} = \sqrt{\text{MSECV}} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(\dot{y}_{c_j} - \hat{\alpha}_{cv_j} - \mathbf{x}_{c_j}\hat{\boldsymbol{\beta}}_{cv_j})^2},$$

where $\hat{\alpha}_{cv_j}$ and $\hat{\boldsymbol{\beta}}_{cv_j}$ are partial least squares regression coefficient estimates of a reduced dataset that does not include the j-th observation.

Apart from RMSEP and RMSECV, other approaches to quantify prediction uncertainty can be classified into one of two types as in Zhang and Garcia-Munoz (2009). One is based on mathematical exploration of prediction error, for instance approximation methods based on the standard expression from multiple regression and linearisation of the estimator; the other is to use re-sampling methods, such as bootstrapping and jackknife.

4.2.2 Ordinary Least Squares Type Mean Squared Error

The earliest form of ordinary least squares type expression for partial least squares prediction variance

$$\mathrm{Var}\,(\hat{\dot{y}}_p) = \frac{\sigma_\epsilon^2}{n} + \mathbf{x}_p \mathrm{Var}\,(\hat{\boldsymbol{\beta}})\mathbf{x}_p' \tag{4.2}$$

was derived directly from the prediction formula under the NIPALS algorithm, Höskuldsson (1988). It assumes the score matrix $\mathbf{T}$ to be fixed, indicating that the variation in the process of choosing latent factors is ignored, so prediction mean squared error has exactly the same form as ordinary least squares regression (See Equation (1.6)). It has been further developed as

$$\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \sigma_\epsilon^2(\frac{1}{n} + h + 1), \tag{4.3}$$

where $h = \mathbf{t}_p(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_p'$ is defined as the leverage. The regression error variance estimate $\hat{\sigma}_{\epsilon_c}^2$ (Equation (4.17)) from the calibration set is used by Zhang and Garcia-Munoz (2009) to replace σ_ϵ^2. Zhang and Garcia-Munoz (2009) also illustrate how to use the SIMPLS algorithm proposed by De Jong (1993) in the calibration of the ordinary least squares type prediction mean squared error, since the SIMPLS algorithm provides a convenient way to calculate normalised scores as well as partial least squares regression coefficients.

4.2.3 Linearisation Based Methods

Linearisation based methods can be achieved by using the linearisation of partial least squares estimators to construct approximate confidence intervals. Like the ordinary least squares prediction mean squared error, it starts from $\mathrm{Var}\,(\hat{y}_p)$, but it differs from the idea of fixed $\mathbf{T}$, since it takes into account the fact that variations do exist in the score matrix $\mathbf{T}$ due to its dependence on the response variable. Denham (1997) is the first paper on the linearisation method, which chooses the point of interest in the linearisation as $\mathbf{y}_{c_0}$. Using the first-order Taylor expansion, the linearisation of the regression coefficient estimator can be approximated as $\hat{\boldsymbol{\beta}}_{\mathbf{y}_c} \approx \hat{\boldsymbol{\beta}}_{\mathbf{y}_{c_0}} + \mathbf{J}(\mathbf{y}_c - \mathbf{y}_{c_0})$, where $\hat{\boldsymbol{\beta}}_{\mathbf{y}_{c_0}}$ are estimated regression coefficients at some point $\mathbf{y}_{c_0}$, and the Jacobian matrix $\mathbf{J}$ $(k \times n)$ is the matrix derivative of $\hat{\boldsymbol{\beta}}$ with respect to $\mathbf{y}_c$ evaluated at $\mathbf{y}_{c_0}$. Ideally, $\mathbf{y}_{c_0}$ approximates to $\mathrm{E}\,(\mathbf{y})_c$, so Denham (1997) uses the fitted values of the centred data as $\mathbf{y}_{c_0}$ and

$$\mathbf{J} = \Big(\frac{\partial \hat{\boldsymbol{\beta}}}{\partial \mathbf{y}_c}\Big)_{\mathbf{y}_{c_0}} = \begin{pmatrix} \frac{\partial \hat{\beta}_1}{\partial y_{c_1}} & \frac{\partial \hat{\beta}_1}{\partial y_{c_2}} & \cdots & \frac{\partial \hat{\beta}_1}{\partial y_{c_n}} \\ \frac{\partial \hat{\beta}_2}{\partial y_{c_1}} & \frac{\partial \hat{\beta}_2}{\partial y_{c_2}} & \cdots & \frac{\partial \hat{\beta}_2}{\partial y_{c_n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \hat{\beta}_k}{\partial y_{c_1}} & \frac{\partial \hat{\beta}_k}{\partial y_{c_2}} & \cdots & \frac{\partial \hat{\beta}_k}{\partial y_{c_n}} \end{pmatrix}_{\mathbf{y}_{c_0}} . \tag{4.4}$$

Hence, the approximate covariance matrix of $\boldsymbol{\beta}$ can be written as

$$\mathrm{Var}\,(\hat{\boldsymbol{\beta}}) = \sigma_\epsilon^2 \mathbf{J}\mathbf{J}'. \tag{4.5}$$

Once the covariance matrix of $\hat{\boldsymbol{\beta}}$ is approximated, the prediction mean squared error can be obtained by plugging Equation (4.5) into the term of $\mathrm{Var}\,(\hat{\boldsymbol{\beta}})$ in ordinary least squares type prediction mean squared error, Equation (4.6).

$$\mathrm{E}\,\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} = \frac{\sigma_\epsilon^2}{n} + \mathbf{x}_p \mathrm{Var}\,(\hat{\boldsymbol{\beta}})\mathbf{x}_p' + \sigma_\epsilon^2. \tag{4.6}$$

The local linearisation approximation gives prediction mean squared error

$$\mathrm{E}\,\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} = \sigma_\epsilon^2(1 + \frac{1}{n} + \mathbf{x}_p \mathbf{J}\mathbf{J}'\mathbf{x}_p'), \tag{4.7}$$

Denham (1997) uses the estimated regression error variance presented in Equation (4.18) of Section 4.2.6 as the substitute for σ_ϵ^2.

The linearisation based method can be categorised into two classes according to the point of interest in the linearisation. One is called the classical methods, referring to these methods based on Denham (1997)'s linearisation method, which regards $\mathbf{y}_{c_0}$ as the point of interest, but different approaches for calculating the covariance matrix of the estimated regression coefficients have been studied.

Denham (1997) considers the variation about the response variable and gives a Jacobian matrix through an inductive algorithm that involves partial least squares regression estimates to be calculated by Helland (1988) (See Algorithm 4.4). The algorithm for calculating the Jacobian matrix, Algorithm 4.5, is presented in Appendix 4.4. It is necessary to calculate a $k^2 \times k$ matrix and a $k^2 \times n$ matrix for each latent variable in the Denham (1997)'s linearisation method. To improve this, Serneels et al. (2004) introduce an efficient algorithm for the Jacobian matrix that decreases the complexity of the largest matrix. The largest matrix in the efficient algorithm is reduced to $k \times k$.

Unlike Denham (1997), which uses an iterative algorithm to calculate $\mathbf{J}$, Phatak et al. (2002) adopts matrix differential calculus techniques inspired by Magnus and Neudecker (1979) and the asymptotic distribution result of the delta method in the calculation of the Jacobian matrix.

The other class is proposed by Romera (2010), which uses a constructed vector as the point of interest. The new point of interest considers the variations in both explanatory and response variables. Romera (2010) applies the first order linearisation at a new point of interest, and it uses the asymptotic distribution result of the delta method in the calculation of the variance of the estimated regression coefficients. The meaning of the work is similar to Phatak et al. (2002)'s, which is a development of the classical linearisation method. However, the mathematics in the paper seems problematic, and there is no simulation result or data analysis to demonstrate the method. We will discuss this method in detail in Chapter 5.

4.2.4 Re-sampling Methods

Linearisation based methods provide analytical solutions to quantify prediction uncertainty, meanwhile re-sampling methods give empirical solutions to this problem. Re-sampling methods such as bootstrapping by objects and bootstrapping by residuals (Efron and Tibshirani (1994), Wehrens and Van Der Linden (1997), Faber (2002)), the jackknife method (Efron and Tibshirani (1994), Faber (2002), Martens and Martens (2000)), and cross-validation (Stone (1974), Martens and Næs (1991), Filzmoser et al. (2009), Xu et al. (2004)), have all been studied to evaluate prediction performance.

1. Bootstrapping by objects

   In bootstrapping by objects, M new data sets are generated by randomly drawing objects with replacement from the calibration set (Faber (2002))

   $$\begin{aligned} (\dot{\mathbf{x}}^b_{c_j}, \dot{y}^b_{c_j}) &= (\dot{\mathbf{x}}_{c,\tau^b_j}, \dot{y}_{c,\tau^b_j}), \qquad j = 1, \ldots, n, \quad b = 1, \ldots, M, \\ \text{where } \tau^b_j &= int[U \cdot n] + 1; \end{aligned} \tag{4.8}$$

   where $int[\cdot]$ denotes the integer part of the associated number and U is a random number generated from the standard uniform distribution. The procedure is repeated M times, where M should be selected large enough to yield precise estimates for the desired variance, then the variance of the estimated partial least squares regression coefficients is

   $$\text{Var}\,(\hat{\boldsymbol{\beta}}) \approx \text{Var}\,(\hat{\boldsymbol{\beta}}^B) = \frac{1}{M-1}\sum_{b=1}^{M}(\hat{\boldsymbol{\beta}}^b - \bar{\boldsymbol{\beta}})(\hat{\boldsymbol{\beta}}^b - \bar{\boldsymbol{\beta}})', \tag{4.9}$$

   where $\hat{\boldsymbol{\beta}}^b$ contains partial least squares regression coefficient estimates for the b-th replicate , and $\bar{\boldsymbol{\beta}}$ denotes the average of all bootstrapped estimated regression coefficients, $\bar{\boldsymbol{\beta}} = \frac{1}{M}\sum_{B=1}^{M}\hat{\boldsymbol{\beta}}^b$. Following this, the prediction mean squared error can be obtained by plugging Equation (4.9) into the ordinary least squares type prediction mean squared error as shown in Equation (4.6).

   $$\text{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \frac{\sigma^2_\epsilon}{n} + \mathbf{x}_p \text{Var}\,(\hat{\boldsymbol{\beta}}^B)\mathbf{x}'_p + \sigma^2_\epsilon. \tag{4.10}$$

Zhang and Garcia-Munoz (2009) use the estimated regression error variance $\hat{\sigma}^2_{\epsilon_c}$ (Equation (4.17) from the calibration set (See Section 4.2.6) as the estimate of σ^2_ϵ.

2. Bootstrapping by residuals

   Bootstrapping by residuals first calculates residuals in the calibration set as

$$\epsilon_j = \frac{\dot{y}_{c_j} - \hat{\dot{y}}_{c_j}}{\sqrt{1-(a+1)/n}}, \qquad j = 1, \ldots, n.$$

   where $\sqrt{1-(a+1)/n}$ is the scaling factor. New residual vectors $\epsilon_{\tau_j^b}$ are generated by randomly drawing residuals with replacement.

$$\begin{aligned} \dot{\mathbf{x}}^b_{c_j} &= \dot{\mathbf{x}}_{c_j} \qquad j = 1, \ldots, n, \quad b = 1, \ldots, M, \\ \dot{y}^b_{c_j} &= \hat{\dot{y}}_{c_j} + \epsilon_{\tau_j^b} \\ &= \hat{\alpha}^b + \mathbf{x}^b_{c_j}\hat{\boldsymbol{\beta}}^b + \epsilon_{\tau_j^b} \end{aligned}$$

   where τ_j^b is defined in Equation (4.8). This procedure is repeated M times. The bootstrap data sets $(\dot{\mathbf{x}}^b_{c_j}, \dot{y}^b_{c_j})$ are used to calculate partial least squares regression coefficients $\hat{\boldsymbol{\beta}}^b$. The variance of estimated regression coefficients is calculated by Equation (4.9), then it is plugged into the ordinary least squares type expression formula. Faber and Bro (2002) have shown that bootstrapping by objects did not give a good result, so we will only use bootstrapping by residuals.

3. The Jackknife Method

   The jackknife generates reduced data sets by deleting objects. The process is the same as the cross-validation introduced in Section 3.2.1. The reduced dataset deleting an observation $\dot{y}_{c_j}$ and its row predictor vector $\dot{\mathbf{x}}_{c_j}$:

$$\begin{aligned} \dot{\mathbf{X}}_{c_{-j}} &= (\dot{\mathbf{x}}_{c_1}, \ldots, \dot{\mathbf{x}}_{c_{j-1}}, \dot{\mathbf{x}}_{c_{j+1}}, \ldots, \dot{\mathbf{x}}_{c_n}) \\ \dot{\mathbf{y}}_{c_{-j}} &= (\dot{y}_{c_1}, \ldots, \dot{y}_{c_{j-1}}, \dot{y}_{c_{j+1}}, \ldots, \dot{y}_{c_n})' \qquad j = 1, \ldots, n. \end{aligned}$$

   Let $\hat{\boldsymbol{\beta}}^{jack_j}$ be the estimated regression coefficients from the reduced dataset that does not include the j-th observation, whilst $\hat{\boldsymbol{\beta}}$ denotes the regression

coefficient estimates using the entire data. The jackknife estimate of the regression coefficient estimate variance can be written as

$$\text{Var}\,(\hat{\boldsymbol{\beta}}^{J}) = \frac{n-1}{n}\sum_{j=1}^{n}(\hat{\boldsymbol{\beta}}^{jack_j} - \bar{\boldsymbol{\beta}})(\hat{\boldsymbol{\beta}}^{jack_j} - \bar{\boldsymbol{\beta}})', \tag{4.11}$$

where $\bar{\boldsymbol{\beta}}$ denotes average regression coefficients of all reduced data sets, $\bar{\boldsymbol{\beta}} = \frac{1}{n}\sum_{j=1}^{n}\hat{\boldsymbol{\beta}}^{jack_j}$. The factor $\frac{n-1}{n}$ corrects for bias (Efron and Tibshirani (1994)). The prediction mean squared error can be obtained by plugging Equation (4.11) into the ordinary least squares type prediction mean squared error (Equation (4.6)). The jackknife can be seen as an approximation to the bootstrap, Efron and Tibshirani (1994). Since it only requires to compute n reduced jackknife data sets, the jackknife method will be cheaper if M is less than 100 or 200 replicates, typically used by the bootstrap method for standard error estimation. On the other hand, as there are only n samples, the jackknife uses limited information to make inference for regression coefficients.

The terminology of jackknife and cross-validation is a little confusing since they apply the same leave-out process to the training set but answer different questions. The jackknife's output is a direct estimate of the variance of the regression coefficient estimates, whilst the cross-validation calculates the root mean squared prediction error of cross-validation (RMSECV) to assess prediction performance. Efron (1982) compared bootstrapping, jackknife and cross-validation through a concept of "excess error", which was originally used to adjust estimation bias. Although the formulae of expected excess error estimate of jackknife and cross-validation are similar, Efron noted that the jackknife involves the predictions of all observations in each reduced dataset, while cross-validation only predicts left-out observations.

Faber (2002) uses simulation and real data analysis to compare these resampling methods in the uncertainty estimation of the estimated regression coefficients, which suggests bootstrapping by residuals performs better than bootstrapping by objects and the jackknife for partial least squares regression. We shall

use bootstrapping by residuals as a representative of the re-sampling methods in the study of Chapter 5.

4.2.5 U-deviation Methods

Chemometrics software Unscrambler originally employed a formula for prediction mean squared error by Harald Martens. It is called U-deviation method.

$$\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \sigma_\epsilon^2\left(\frac{1}{n} + \frac{1}{2}h + \frac{V_{x_p}}{2V_{\mathbf{X}_{t,tot}}}\right), \tag{4.12}$$

where σ_ϵ^2 is replaced by $\hat{\sigma}_\epsilon^2 = \mathrm{MSEP}$, Equation (1.8); The leverage is defined as $h = \mathbf{t}_p(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_p'$; $V_{x_p} = \frac{1}{k-a}\sum_{s=1}^{k} e_{p_s}^2$ is the residual variance of the predictor $\dot{\mathbf{x}}_p$ after modelling, where a is the number of factors and e_{p_s} are the elements in the x-residual vector $\mathbf{e}_p = \mathbf{x}_p - \mathbf{x}_p\mathbf{V}\hat{\mathbf{q}}$; $V_{\mathbf{X}_{t,tot}} = \frac{1}{l(k-a)}\sum_{j=1}^{l}\sum_{s=1}^{k} e_{t_{js}}^2$ is an average residual variance of the tuning set, where $e_{t_{js}}$ are the elements of x-residual matrix in the tuning set $\mathbf{E}_{t_j}$. If the $\mathbf{x}_p$ of the observation to be predicted is similar to the observations in the tuning set $\frac{V_{x_p}}{V_{\mathbf{X}_{t,tot}}} \approx 1$.

On the basis of $\hat{\dot{y}}_p = \bar{\dot{y}} + \mathbf{t}_p\hat{\mathbf{q}}$, the U-deviation method formula Equation (4.12) seems not convincing because the variance of the product of $\mathbf{t}_p$ and $\hat{\mathbf{q}}'$ is taken as an average, as shown:

$$\mathrm{Var}\left(\mathbf{t}_p\hat{\mathbf{q}}'\right) = \frac{1}{2}\mathbf{t}_p\mathrm{Var}\left(\hat{\mathbf{q}}\right)\mathbf{t}_p' + \frac{1}{2}\hat{\mathbf{q}}\mathrm{Var}\left(\mathbf{t}_p\right)\hat{\mathbf{q}}' = \frac{1}{2}h + \frac{V_{x_p}}{2V_{\mathbf{X}_{t,tot}}}. \tag{4.13}$$

The use of $\frac{1}{2}$ is incorrect. It should simply be a sum according to the derivative sum rule. Nor is it quite obvious where the second term involving the x-residual comes from.

From Equation (4.12), we can see the leverage h describes the distance between the projections of the predictor to the centre of the new factor space. This has also been considered in the ordinary least squares type expression. The extra term, the ratio of V_{x_p} and $V_{\mathbf{X}_{t,tot}}$ links the prediction mean squared error to the x-residuals. Intuitively, it seems reasonable that new observations with large x-residual may predict less well, but it is not clear why this form of dependence should be the correct one.

Fernández Pierna et al. (2003) points out U-deviation formula is problematic. A relatively large value of the ratio $\frac{V_{x_p}}{V_{\mathbf{X}_{t,tot}}}$ in Equation (4.12) could be understood as a poor model fit because large x-residuals (the numerator), in comparison with the tuning set (the denominator), results in a relatively large contribution of the ratio to prediction mean squared error. The size of x-residuals does not relate to prediction mean squared error in an obvious way. In the 'worst' case, the interference of large measurement errors could change the score $\mathbf{t}_p$, but leads to normal x-residuals. The predicted value $\hat{\dot{y}}_p$ has large variation since it is directly linked to $\mathbf{t}_p$, but we could not see a big prediction mean squared error because of the normal x-residuals, then the U-deviation prediction mean squared error is over-optimistic. In the 'best' case, large measurement errors directly go into x-residuals. This would inflate prediction mean squared error, but the score $\mathbf{t}_p$ remains the same if the score $\mathbf{t}_p$ is orthogonal to the error in $\mathbf{x}_p$, so the predicted value $\hat{\dot{y}}_p$ would not be affected by the interference of measurement errors. In this case, the U-deviation prediction mean squared error is pessimistic.

De Vries and J.F. Ter Braak (1995) points out that the U-deviation formula (Equation (4.12)) underestimates partial least squares prediction uncertainty and suggests an ad-hoc fix where the number of factors is used. The improved formula below has been confirmed by Høy et al. (1998),

$$\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = 2\sigma_\epsilon^2(1 - \frac{a+1}{n})(\frac{1}{n} + \frac{1}{2}h + \frac{V_{x_p}}{2V_{\mathbf{X}_{t,tot}}}). \tag{4.14}$$

However, the adjustment does not correct the mistakes made by the U-deviation method.

To improve on Marten's approach by taking the variation about $\dot{\mathbf{y}}_c$ into account, Faber and Kowalski (1996) proposes an ordinary least squares type approximation of prediction mean squared error that considers all measurement errors in both explanatory and response variables under the general errors-in-variables (EIV) models: $\dot{\mathbf{y}}_c = \tilde{\dot{\mathbf{y}}}_c + \Delta\dot{\mathbf{y}}_c$ and $\dot{\mathbf{X}}_c = \tilde{\dot{\mathbf{X}}}_c + \Delta\dot{\mathbf{X}}_c$. $\dot{\mathbf{y}}_c$ is the measured response variable, $\tilde{\dot{\mathbf{y}}}_c$ is the true response variable, and $\Delta\dot{\mathbf{y}}_c$ contains the measurement errors in $\dot{\mathbf{y}}_c$. $\dot{\mathbf{X}}_c$ is the measured explanatory variable matrix, $\tilde{\dot{\mathbf{X}}}_c$ is the true explanatory variable matrix, $\Delta\dot{\mathbf{X}}_c$ contains the measurement errors in $\dot{\mathbf{X}}_c$. Prediction mean squared

error can be expressed as

$$\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} \approx (\frac{1}{n} + h)\{\sigma_\epsilon^2 + \sigma^2_{\Delta\dot{\mathbf{y}}_c} + \|\boldsymbol{\beta}\|_2^2 \sigma^2_{\Delta\dot{\mathbf{X}}_c}\} + \sigma_\epsilon^2 + \|\boldsymbol{\beta}\|_2^2 \sigma^2_{\Delta\dot{\mathbf{x}}_p}, \qquad (4.15)$$

where h is defined as the same as Equation (4.3), and $\|\cdot\|_2$ denotes Euclidean vector norm. If we neglect measurement errors, $\sigma^2_{\Delta\dot{\mathbf{y}}_c} = 0$, $\sigma^2_{\Delta\dot{\mathbf{X}}_c} = 0$, and assume the variance of regression error in the prediction set equals to that in the calibration $\sigma^2_{\Delta\dot{\mathbf{x}}_p} = \sigma^2_{\Delta\dot{\mathbf{X}}_c}$, the special case of Equation (4.15) would be identical to Equation (4.3). Faber and Bro (2002) have further simplified Equation (4.15) to

$$\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} \approx (\frac{1}{n} + h + 1)\mathrm{MSEC} - \sigma^2_{\Delta\dot{\mathbf{y}}_c}. \qquad (4.16)$$

One possible way to find the estimates of $\sigma^2_{\Delta\dot{\mathbf{y}}_c}$, $\sigma^2_{\Delta\dot{\mathbf{X}}_c}$ and their degrees of freedom, is from replicate measurements. Faber and Kowalski (1997) gave an alternative approach for this.

4.2.6 Regression Error Variance Estimates and Degrees of Freedom

From the previous sections, we can see that the regression error variance σ_ϵ^2 plays an important role in these prediction mean squared error formulae. The estimation of the regression error variance is often associated with the degrees of freedom. We will focus on the two topics in this section.

In the literature, the estimated regression error variance from the calibration set is often used. Similarly to multiple linear regression, the estimate from the calibration set can be written as

$$\hat{\sigma}^2_{\epsilon_c} = \mathrm{MSEC} = \frac{1}{n - a - 1} \sum_{j=1}^{n} (\dot{y}_{c_j} - \hat{\alpha} - \mathbf{x}_{c_j}\hat{\boldsymbol{\beta}})^2. \qquad (4.17)$$

where a is the number of latent factors. Here $n - a - 1$ is a simple estimate of degrees of freedom. Although the actual degrees of freedom is unknown, the use of the number of factors a is certain to overestimate the degrees of freedom, hence Equation (4.17) underestimates regression error variance. This estimate of

regression variance has been suggested to used in the ordinary least squares type expression, and bootstrapping prediction mean squared error formulae.

Denham (1997) also employs an estimated regression error variance from the calibration set, but it is somewhat different. Denote the residual $\hat{\boldsymbol{\epsilon}} = \dot{\mathbf{y}}_c - \hat{\dot{\mathbf{y}}}_c$, and its first derivative with respect to $\mathbf{y}_c$ evaluated at $\mathbf{y}_{c_0}$ as $\mathring{\hat{\boldsymbol{\epsilon}}}$. The mathematical form of $\mathring{\hat{\boldsymbol{\epsilon}}}$ is presented in Appendix 4.4.

$$\hat{\sigma}^2_{\epsilon_d} = \frac{\hat{\boldsymbol{\epsilon}}'\hat{\boldsymbol{\epsilon}} - ||\hat{\boldsymbol{\epsilon}} - \mathring{\hat{\boldsymbol{\epsilon}}}\hat{\boldsymbol{\epsilon}}||^2}{\mathrm{tr}(\mathring{\hat{\boldsymbol{\epsilon}}}'\mathring{\hat{\boldsymbol{\epsilon}}})}, \tag{4.18}$$

where the degrees of freedom is estimated as the trace of $\mathring{\hat{\boldsymbol{\epsilon}}}'\mathring{\hat{\boldsymbol{\epsilon}}}$. There are some more works about the degrees of freedom, for example Faber and Kowalski (1997), Van Der Voet (1999), and Ye (1998).

An alternative to estimate the regression error variance from the calibration set is to use the tuning set. The U-deviation method use the MSEP, Equation (1.8), as the estimate of regression error variance. The leave-one-out cross-validation can be used as the same way as the tuning set to calculate the estimated regression error variance, Baumann and Stiefl (2004).

We propose an estimated regression error variance from the tuning set. The idea comes from the empirical result MSEP, but it avoids the estimation of the degrees of freedom, so the mathematical process is not complicated. It also captures the nature that there exists bias in partial least squares regression, so the estimated regression error variance is not only an estimate of the variation about the regression, but is also formative for the estimation of the bias. The new estimated regression error variance can be written as

$$\hat{\sigma}^2_{\epsilon_t} = \frac{\frac{1}{n_t}\sum_{j=1}^{n_t}(\dot{y}_{t_j} - \hat{\dot{y}}_{t_j})^2}{\frac{1}{n_t} + \frac{1}{n_t}\sum_{j=1}^{n_t} h_{t_j} + 1}, \tag{4.19}$$

where the leverage of the j-th observation in the tuning set is defined as $h_{t_j} = \mathbf{t}_{t_j}(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}'_{t_j}$. n_t is the number of observation in the tuning set. Using the empirical prediction mean squared error formula in this way should at least ensure average prediction mean squared error is roughly correct, but the dependence on the predictor via the leverage in Equation (4.3) may not be correct.

In Chapter 5, we will use the estimated regression error variances from the calibration set, Equation (4.17) and from the tuning set, Equation (4.19), to compare different prediction mean squared error formulae.

4.3 Summary

The ordinary least squares type prediction mean squared error only considers the variations about explanatory variables via the distance between the projection of the predictor to the projection of the centred explanatory variables. The U-deviation method adds x-residuals into the ordinary least squares type expression. Although the idea that the distance between the predictor and its projection on the new factor space may be relevant to the prediction error is a sensible one, the form of the dependence in this method seems arbitrary.

Denham (1997)'s linearisation method studies estimated regression coefficients with small change of the response variable. Re-sampling methods includes the information about the response variable during the re-sampling process.

Romera (2010)'s idea takes into account the variations about explanatory and response variables together, which is similar to Faber and Kowalski (1996), the improved U-deviation method that considers the measurement errors in both explanatory and response variables.

Since there are mistakes in the U-deviation method, we will drop it from the comparison. We will use simulation study and real data analysis to compare the ordinary least squares prediction mean squared error, Denham (1997)'s linearisation prediction mean squared error, re-sampling by residuals prediction mean squared error, and a new linearisation method built on Romera (2010)'s idea in Chapter 5.

4.4 Appendix

Algorithm 4.5. Jacobian Matrix Algorithm Proposed in Denham (1997)'s Linearisation Method

The Jacobian matrix algorithm is designed based on the orthogonal loadings algorithm (Algorithm 4.4), which calculates $\mathbf{w}_i$, $\tilde{\mathbf{w}}_i$, and $\mathbf{H}_i$. For $i = 1, \cdots, a$,

- $$\frac{\partial \mathbf{H}_i}{\partial \mathbf{y}_c} = \mathbf{0}\,(k^2 \times n),$$

- $$\begin{aligned}\frac{\partial \mathbf{w}_i}{\partial \mathbf{y}_c} &= (\mathbf{I} - \mathbf{S}_{xx}\mathbf{H}_{i-1})\mathbf{X}_c'(\mathbf{I} - \bar{\mathbf{J}}) \\ &\quad -\mathbf{S}_{xx}(\mathbf{s}_{xy}' \otimes \mathbf{I}_k)\frac{\partial \mathbf{H}_{i-1}}{\partial \mathbf{y}_c},\end{aligned}$$

- $$\begin{aligned}\frac{\partial \tilde{\mathbf{w}}_i}{\partial \mathbf{y}_c} &= (\mathbf{I} - \mathbf{H}_{i-1}\mathbf{S}_{xx})\frac{\partial \mathbf{w}_i}{\partial \mathbf{y}_c} \\ &\quad -(\mathbf{w}_i'\mathbf{S}_{xx} \otimes \mathbf{I}_k)\frac{\partial \mathbf{H}_{i-1}}{\partial \mathbf{y}_c},\end{aligned}$$

- $$\begin{aligned}\frac{\partial \mathbf{H}_i}{\partial \mathbf{y}_c} &= \frac{\partial \mathbf{H}_{i-1}}{\partial \mathbf{y}_c} + \frac{\partial \tilde{\mathbf{w}}_i}{\partial \mathbf{y}_c} \otimes \frac{\tilde{\mathbf{w}}_i}{\tilde{\mathbf{w}}_i'\mathbf{S}_{xx}\tilde{\mathbf{w}}_i} \\ &\quad + \frac{\tilde{\mathbf{w}}_i}{\tilde{\mathbf{w}}_i'\mathbf{S}_{xx}\tilde{\mathbf{w}}_i} \otimes \frac{\partial \tilde{\mathbf{w}}_i}{\partial \mathbf{y}_c} \\ &\quad - \frac{2\tilde{\mathbf{w}}_i \otimes \tilde{\mathbf{w}}_i}{(\tilde{\mathbf{w}}_i'\mathbf{S}_{xx}\tilde{\mathbf{w}}_i)^2}\tilde{\mathbf{w}}_i'\mathbf{S}_{xx}\frac{\partial \tilde{\mathbf{w}}_i}{\partial \mathbf{y}_c},\end{aligned}$$

- $$\frac{\partial \boldsymbol{\beta}_i}{\partial \mathbf{y}_c} = \mathbf{H}_i\mathbf{X}_c'(\mathbf{I} - \bar{\mathbf{J}}) + (\mathbf{s}_{xy}' \otimes \mathbf{I}_k)\frac{\partial \mathbf{H}_i}{\partial \mathbf{y}_c},$$

where $\bar{\mathbf{J}}$ is an $n \times n$ matrix whose elements are all equal to $1/n$. The Jacobian matrix in Equation (4.7)

$$\mathbf{J} = \frac{\partial \boldsymbol{\beta}_a}{\partial \mathbf{y}_c}.$$

In the estimation of regression error variance, the first derivative of the residual with respect to $\mathbf{y}_c$ evaluated at $\mathbf{y}_{c0}$ can be written as $\mathring{\boldsymbol{\epsilon}} = (\mathbf{I} - \bar{\mathbf{J}})(\mathbf{I} - \mathbf{X}_c\frac{\partial \hat{\boldsymbol{\beta}}_a}{\partial \mathbf{y}_c})$.

Chapter 5

A Modified Partial Least Squares Linearisation Method

This chapter presents an original contribution of the thesis. It focuses on a new local linearisation method proposed recently (Romera (2010)), which is found to be problematic. Following Romera (2010)'s idea, two algorithms to implement the new linearisation method have been developed from scratch. We use simulated and real data analyses to compare the new method with other existing partial least squares prediction uncertainty approaches, such as the ordinary least squares type prediction mean squared error, Denham (1997)'s linearisation method, and bootstrapping by residuals.

- Section 5.1 gives the background knowledge of building the new linearisation method.
- Section 5.2 introduces the new linearisation method, and presents the mathematical derivation of the new algorithm.
- In Section 5.3 constructs an alternative algorithm of the new linearisation method embedded with bootstrapping.
- Section 5.4 summarises different prediction variances being discussed in the simulation and real data analysis.

- Section 5.5 applies the new linearisation method on the simulated data in comparison with standard approaches.
- Section 5.6 carries out a real data analysis with the help of random data splitting. To help understand the real data analysis, a series of simulations are run to investigate the use of random data splitting in partial least squares regression.

5.1 Background

Before starting the theory of the new linearisation method, we introduce some definitions.

Definition 5.1. Assume $\mathbf{g}$ to be an $l \times 1$ column vector, and $\mathbf{v}$ to be an $r \times 1$ column vector. The derivative $\partial \mathbf{g}/\partial \mathbf{v}$ is an $r \times l$ matrix with the (i, j)-th element defined as $\partial g_i/\partial v_j$.

Definition 5.2. Let *vecut* denote an operator that gives a column vector whose elements are taken in order along rows including the diagonal elements from the upper triangular part of a symmetric matrix.

Definition 5.3. Let *diag* denote an operator that extracts the diagonal terms from a symmetric matrix as a column vector.

The multiple linear regression model of a single response variable can be written as

$$\dot{\mathbf{y}}_c = \beta_0 + \dot{\mathbf{X}}_c \boldsymbol{\beta} + \boldsymbol{\epsilon}_c,$$

where $\dot{\mathbf{y}}_c$ $(n \times 1)$ is the response variable, $\dot{\mathbf{X}}_c$ $(n \times k)$ denotes the explanatory variables, β_0 and $\boldsymbol{\beta}$ $(k \times 1)$ are regression coefficients, and $\boldsymbol{\epsilon}_c$ $(n \times 1)$ is the error term that is independent and identically normally distributed with mean 0 and variance σ_ϵ^2. The centred predictor $\mathbf{x}_p = \dot{\mathbf{x}}_p - \bar{\dot{\mathbf{x}}}$, where $\bar{\dot{\mathbf{x}}}$ is the mean of explanatory variables.

Section 4.2.3 gives an introduction to the linearisation approach, one of the important methods to quantify prediction uncertainty in the literature. As a

typical example of the classical linearisation approaches, Denham (1997) constructs approximate prediction intervals via a local linear approximation, which takes into account the fact that the variation exists in generating the scores $\mathbf{T}$. As shown in Equation (4.7), the prediction mean squared error proposed by Denham (1997) can be expressed as

$$\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \sigma_\epsilon^2\{1 + \frac{1}{n} + \mathbf{x}_p\mathbf{J}\mathbf{J}'\mathbf{x}_p'\}.$$

Romera (2010) picks up the basic ideas used in Denham (1997) and Phatak et al. (2002), but carries out the linearisation at a different local point. Romera (2010) criticises the fact that δ-method based approach depends on taking the fitted value as its initial estimate. Since prediction uncertainty is unknown, it is unclear how prediction uncertainty of the fitted value would affect the quantification of prediction uncertainty in partial least squares regression. To avoid using partial least squares regression on the data twice, Romera (2010) chooses to implement the linearisation around a vector that consists of the covariance between explanatory and response variables, and the variance of explanatory variables. Different from the classic approach, Romera (2010) constructs a Jacobian matrix that does not involve the use of the Kronecker product. However, there is no simulation or real data analysis given by Romera (2010). We find the proposed algorithm is incomplete, and some parts of the algorithm need a better presentation.

Romera (2010) constructs a new column vector with the dimension of $k(k+3)/2 \times 1$, $\mathbf{b} = \begin{pmatrix} n\boldsymbol{\gamma} \\ vecut\,(n\boldsymbol{\Sigma}) \end{pmatrix}$, where $\boldsymbol{\gamma}$ denotes the covariance between explanatory variables and the response variable, $\boldsymbol{\gamma} = (\ \gamma_1 \quad \gamma_2 \quad \cdots \quad \gamma_k\ )'$. The use of the new vector $\mathbf{b}$ considers the variations in both of explanatory variables and the response variable. It also defines a local point of interest $\mathbf{b}_0 = \begin{pmatrix} \mathbf{s}_{xy} \\ vecut\,(\mathbf{S}_{xx}) \end{pmatrix}$, which is the sample version of $\mathbf{b}$. $\mathbf{b}_0$ is a function of the sample covariance of explanatory variables and the response variable and the sample variance of explanatory variables, which gives initial information about the dataset.

Romera (2010) explores the dependence of regression coefficients on $\mathbf{b}$ via y-

loadings, because in the last step of partial least squares regression, the y-loadings q_i is estimated by regressing $\mathbf{y}_c$ on the scores $\mathbf{t}_i$ as shown in Algorithm 4.2 where $\hat{q}_i = \mathbf{y}_c'\mathbf{t}_i/(\mathbf{t}_i'\mathbf{t}_i)$. The estimated y-loadings around some point $\mathbf{b}_0$ can be expanded according to the first-order Taylor expansion

$$\hat{\mathbf{q}}_{\mathbf{b}} \approx \hat{\mathbf{q}}_{\mathbf{b}_0} + \mathbf{J}(\mathbf{b} - \mathbf{b}_0).$$

The variance of explanatory variables is denoted by $\mathbf{\Sigma} = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1k} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{k1} & \sigma_{k2} & \cdots & \sigma_{k1} \end{pmatrix}$. The approximate variance of the estimated y-loadings $\text{Var}(\hat{\mathbf{q}}) \approx \mathbf{J}\text{Var}(\mathbf{b}_0)\mathbf{J}'$, where the Jacobian matrix $\mathbf{J}$ $(a \times k(k+3)/2)$ is the first derivative of $\hat{\mathbf{q}}$ with respect to $\mathbf{b}$ evaluated at $\mathbf{b}_0$, $\mathbf{J} = \left(\partial\hat{\mathbf{q}}/\partial\mathbf{b}\right)_{\mathbf{b}_0}$. The initial starting point in the PLS algorithm is chosen as $\mathbf{b}_1 = \mathbf{b}_0$.

Following the ordinary least squares prediction mean squared error (Equation (4.2)) and $\hat{\boldsymbol{\beta}} = \mathbf{W}\hat{\mathbf{q}}$ which gives $\text{Var}(\hat{\boldsymbol{\beta}}) = \mathbf{W}\text{Var}(\hat{\mathbf{q}})\mathbf{W}'$, Romera (2010) proposes mean squared prediction error

$$\text{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \sigma_\epsilon^2(1 + \frac{1}{n}) + \mathbf{x}_p\mathbf{W}\mathbf{J}\text{Var}(\mathbf{b}_0)\mathbf{J}'\mathbf{W}'\mathbf{x}_p'. \tag{5.1}$$

Both of Denham (1997) and Romera (2010)'s works are built on top of the ordinary least squares prediction mean squared error. Denham (1997) develop the classical linearisation method based on the orthogonal loadings algorithm (Algorithm 4.4). Romera (2010) develops the new method following the orthogonal scores algorithm (Algorithm 4.2). As mentioned in Algorithm 4.4, the orthogonal scores algorithm and the the orthogonal loadings algorithm are proved to be equivalent by Helland (1990), so it would be interesting and meaningful to compare the methods by Denham (1997) and Romera (2010). Denham (1997) obtains $\text{Var}(\hat{\boldsymbol{\beta}})$ directly from its first order Taylor expansion, while Romera (2010) estimates $\text{Var}(\hat{\boldsymbol{\beta}})$ via the first order Taylor expansion of $\text{Var}(\hat{\mathbf{q}})$. The problem with Romera (2010)'s new idea exists in Equation (5.1). As shown in Equation (4.1), for the orthogonal scores algorithm, the estimated regression coefficients can be

calculated by $\hat{\boldsymbol{\beta}} = \mathbf{W}(\hat{\mathbf{P}}'\mathbf{W})^{-1}\hat{\mathbf{q}}$, but not $\hat{\boldsymbol{\beta}} = \mathbf{W}\hat{\mathbf{q}}$, which is the result of the orthogonal loadings algorithm. Although the two formulae have the same column space, the weight matrix $\mathbf{W}$ and the loading vector $\hat{\mathbf{q}}$ are not the same in the two algorithms. Moreover, the local point $\mathbf{b}$ is defined to contain the covariance of explanatory and response variables, and $\mathbf{w}_1$ is defined as a function of sample covariance of explanatory and response variables, hence the weight matrix $\mathbf{W}$ is correlated with $\mathbf{b}$, so the prediction mean squared error cannot be simply obtained by taking $\mathbf{W}(\hat{\mathbf{P}}\mathbf{W})^{-1}$ as a constant.

5.2 New Linearisation Method Theory

To modify Romera (2010)'s method, a sensible approach is to consider the estimated regression coefficients as a whole with respect to the local point $\mathbf{b}_0$. The prediction mean squared error can be written as

$$\mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} = \sigma_\epsilon^2(1 + \frac{1}{n}) + \mathbf{x}_p\big(\frac{\partial\hat{\boldsymbol{\beta}}}{\partial\mathbf{b}}\big)_{\mathbf{b}_0}\mathrm{Var}\,(\mathbf{b}_0)\big(\frac{\partial\hat{\boldsymbol{\beta}}}{\partial\mathbf{b}}\big)'_{\mathbf{b}_0}\mathbf{x}'_p. \tag{5.2}$$

To pursue computational convenience, we take the calculation of $(\partial\hat{\beta}_l/\partial\mathbf{b})_{\mathbf{b}_0}$ for example, where $\hat{\beta}_l$ is the l-th element of $\hat{\boldsymbol{\beta}}$, $(l = 1, \ldots, k)$. Let $\tilde{\mathbf{w}}_l$ denote the l-th row vector of the weight matrix $\mathbf{W}$, where $\tilde{\mathbf{w}}_l = (\ w_{1l} \quad w_{2l} \quad \cdots \quad w_{al}\ )$, and let $\tilde{\mathbf{R}} = (\hat{\mathbf{P}}'\mathbf{W})^{-1}$. Then

$$\begin{aligned} \hat{\beta}_l &= \tilde{\mathbf{w}}_l\tilde{\mathbf{R}}\hat{\mathbf{q}}. \\ \big(\frac{\partial\hat{\beta}_l}{\partial\mathbf{b}}\big)_{\mathbf{b}_0} &= \tilde{\mathbf{w}}_l\tilde{\mathbf{R}}\big(\frac{\partial\hat{\mathbf{q}}}{\partial\mathbf{b}}\big)_{\mathbf{b}_0} + \hat{\mathbf{q}}'\big(\frac{\partial\tilde{\mathbf{w}}_l\tilde{\mathbf{R}}}{\partial\mathbf{b}}\big)_{\mathbf{b}_0}, \end{aligned}$$

and $(\partial\hat{\boldsymbol{\beta}}/\partial\mathbf{b})_{\mathbf{b}_0}$ can be completed by running the algorithm repeatedly.

Section 5.2.1 shows how to obtain the asymptotic result of $\mathrm{Var}\,(\mathbf{b}_0)$ used in Equations (5.1) and (5.2). Section 5.2.2 gives the calculations of $(\partial\hat{\mathbf{q}}/\partial\mathbf{b})_{\mathbf{b}_0}$. In Section 5.2.3 $(\partial\tilde{\mathbf{w}}_l\tilde{\mathbf{R}}/\partial\mathbf{b})_{\mathbf{b}_0}$ is calculated.

5.2.1 The Asymptotic Distribution of $\mathrm{Var}(\mathbf{b}_0)$

Let us consider a new matrix including all the data,

$$\mathbf{C} = \begin{pmatrix} y_{c_1} & x_{c_{11}} & \cdots & x_{c_{1k}} \\ y_{c_2} & x_{c_{21}} & \cdots & x_{c_{2k}} \\ \vdots & \vdots & \ddots & \vdots \\ y_{c_n} & x_{c_{n1}} & \cdots & x_{c_{nk}} \end{pmatrix}.$$

Its covariance matrix can be written as $\mathbf{\Psi} = \begin{pmatrix} \sigma_y^2 & \boldsymbol{\gamma}' \\ \boldsymbol{\gamma} & \mathbf{\Sigma} \end{pmatrix}$, and its sum of squares $\mathbf{G} = \mathbf{C}'\mathbf{C}$. If explanatory variables are assumed to be multivariate normally distributed then $\mathbf{C} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Psi})$, and $\mathbf{G}$ has a Wishart distribution. Magnus and Neudecker (1979) gives the variance of the column stacked vector $\mathbf{G}$,

$$\mathrm{Var}\{vec(\mathbf{G})\} = n(\mathbf{I}_{(1+k)^2} + \mathbf{K})(\mathbf{\Psi} \otimes \mathbf{\Psi}),$$

where $\mathbf{K}$ is a commutation matrix $\mathbf{K} = \sum_{i=1}^{1+k}\sum_{j=1}^{1+k} \mathbf{M}_{ij} \otimes \mathbf{M}'_{ij}$. $\mathbf{M}_{ij}$ is a $(1+k) \times (1+k)$ square matrix with the (i,j)-th element equal to 1 and all other elements being zero. $\mathrm{Var}(\mathbf{b}_0)$ can be obtained by selecting relevant elements from $\mathrm{Var}\{vec(\mathbf{G})\}$, because all elements in $\mathbf{b}_0$ also belong to $vec(\mathbf{G})$.

5.2.2 $\partial\hat{\mathbf{q}}/\partial\mathbf{b}$

To calculate $\partial\hat{\mathbf{q}}/\partial\mathbf{b}$, Romera (2010) continues exploring orthogonal scores algorithm. For each iteration, let us define the sum of squares as below.

$$\begin{aligned} \mathbf{s}_i &= \mathbf{X}'_{c_i}\mathbf{y}_{c_i} = \mathbf{E}'_{i-1}\mathbf{f}_{i-1} = \mathbf{w}_i. \\ \mathbf{S}_i &= \mathbf{E}'_i\mathbf{E}_i = \begin{pmatrix} S_{i_{11}} & S_{i_{12}} & \cdots & S_{i_{1k}} \\ S_{i_{21}} & S_{i_{22}} & \cdots & S_{i_{2k}} \\ \vdots & \vdots & \ddots & \vdots \\ S_{i_{k1}} & S_{i_{k2}} & \cdots & S_{i_{kk}} \end{pmatrix}. \\ \mathbf{b}_i &= \begin{pmatrix} w_{i1} & \cdots & w_{ik} & S_{i_{11}} & S_{i_{12}} & \cdots & S_{i_{kk}} \end{pmatrix}'. \end{aligned}$$

The following properties are useful in the calculation of the Jacobian matrix.

$$\begin{aligned}
\mathbf{A}_i &= \mathbf{I} - \mathbf{S}_i\mathbf{w}_i\mathbf{w}_i'/(\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i). \\
\mathbf{X}_{c_{i+1}} &= \mathbf{X}_{c_i} - \mathbf{t}_i\mathbf{p}_i' = \mathbf{X}_{c_i}(\mathbf{I} - \frac{\mathbf{w}_i\mathbf{w}_i'\mathbf{S}_i}{\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i}) = \mathbf{X}_{c_i}\mathbf{A}_i'. \\
\mathbf{w}_{i+1} &= \mathbf{X}_{c_{i+1}}'\mathbf{y}_c = \mathbf{A}_i\mathbf{X}_{c_i}'\mathbf{y}_c = \mathbf{A}_i\mathbf{w}_i. \qquad (5.3) \\
\mathbf{S}_i &= \mathbf{X}_{c_{i+1}}'\mathbf{X}_{c_{i+1}} = \mathbf{A}_i\mathbf{S}_i\mathbf{A}_i'. \qquad (5.4)
\end{aligned}$$

Since the orthogonal scores algorithm starts at the original centred data $(\mathbf{X}_c, \mathbf{y}_c)$, and $\mathbf{b}_0$ is also defined by the sum of squares, $\mathbf{b}_0 = \mathbf{b}_1$. At each iteration, according to the chain rule we have

$$(\frac{\partial\hat{q}_i}{\partial\mathbf{b}})_{\mathbf{b}_0} = \frac{\partial\hat{q}_i}{\partial\mathbf{b}_i}\frac{\partial\mathbf{b}_i}{\partial\mathbf{b}_{i-1}}\frac{\partial\mathbf{b}_{i-1}}{\partial\mathbf{b}_{i-2}}\cdots\frac{\partial\mathbf{b}_3}{\partial\mathbf{b}_2}\frac{\partial\mathbf{b}_2}{\partial\mathbf{b}_1}. \qquad (5.5)$$

Section 5.2.2.1 and Section 5.2.2.2 will continue working on the calculations of $\partial\hat{q}_i/\partial\mathbf{b}_i$ and $\partial\mathbf{b}_{i+1}/\partial\mathbf{b}_i$.

5.2.2.1 Calculate $\partial\hat{q}_i/\partial\mathbf{b}_i$

$$\begin{aligned}
\frac{\partial\hat{q}_i}{\partial\mathbf{b}_i} &= \begin{pmatrix} \frac{\partial\hat{q}_i}{\partial\mathbf{w}_i} & \frac{\partial\hat{q}_i}{\partial vecut(\mathbf{S}_i)} \end{pmatrix} = \begin{pmatrix} \frac{\partial\hat{q}_i}{\partial\mathbf{w}_i} & vecut(\frac{\partial\hat{q}_i}{\partial\mathbf{S}_i}) \end{pmatrix}. \\
\frac{\partial\hat{q}_i}{\partial\mathbf{w}_i} &= \frac{\partial}{\partial\mathbf{w}_i}(\frac{\mathbf{w}_i'\mathbf{w}_i}{\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i}) \\
&= \frac{\partial}{\partial\mathbf{w}_i}(\mathbf{w}_i'\mathbf{w}_i)\frac{1}{\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i} + \mathbf{w}_i'\mathbf{w}_i\frac{\partial}{\partial\mathbf{w}_i}(\frac{1}{\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i}) \\
&= \frac{2\mathbf{w}_i'}{\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i} - \frac{\mathbf{w}_i'\mathbf{w}_i}{(\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i)^2}\frac{\partial\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i}{\partial\mathbf{w}_i} \\
&= \frac{2\mathbf{w}_i}{\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i} - \frac{\mathbf{w}_i'\mathbf{w}_i}{(\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i)^2}2\mathbf{w}_i'\mathbf{S}_i. \qquad (5.6) \\
\frac{\partial\hat{q}_i}{\partial\mathbf{S}_i} &= -\frac{\mathbf{w}_i'\mathbf{w}_i}{(\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i)^2}\frac{\partial\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i}{\partial\mathbf{S}_i} \\
&= -\frac{\mathbf{w}_i'\mathbf{w}_i}{(\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i)^2}\{2\mathbf{w}_i\mathbf{w}_i' - diag(\mathbf{w}_i\mathbf{w}_i')'\mathbf{I}\}. \qquad (5.7)
\end{aligned}$$

5.2.2.2 Calculate $\partial \mathbf{b}_{i+1}/\partial \mathbf{b}_i$

For the i-th iteration, the factor $\partial \mathbf{b}_{i+1}/\partial \mathbf{b}_i$ used in the chain rule in Equation (5.5) can be decomposed into four blocks:

$$\frac{\partial \mathbf{b}_{i+1}}{\partial \mathbf{b}_i} = \begin{pmatrix} ① & ② \\ ③ & ④ \end{pmatrix}.$$

Block ① is a $k \times k$ matrix.

$$\begin{aligned} ① \quad & \frac{\partial \mathbf{w}_{i+1}}{\partial \mathbf{w}_i} = \frac{\partial \mathbf{A}_i \mathbf{w}_i}{\partial \mathbf{w}_i} \quad \text{using Equation (5.3)} \\ = \quad & \frac{\partial}{\partial \mathbf{w}_i}(\mathbf{w}_i - \frac{\mathbf{S}_i \mathbf{w}_i \mathbf{w}_i' \mathbf{w}_i}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i}) \\ = \quad & \mathbf{I} - \mathbf{S}_i q_i - \mathbf{S}_i \mathbf{w}_i \frac{\partial \hat{q}_i}{\partial \mathbf{w}_i}, \end{aligned}$$

where $\partial \hat{q}_i/\partial \mathbf{w}_i$ is calculated in Equation (5.6).

Block ② is a $k \times \frac{k(k+1)}{2}$ matrix.

$$\begin{aligned} ② \quad & \frac{\partial \mathbf{w}_{i+1}}{\partial vecut(\mathbf{S}_i)} = \frac{\partial}{\partial vecut(\mathbf{S}_i)}(-\frac{\mathbf{S}_i \mathbf{w}_i \mathbf{w}_i' \mathbf{S}_i}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i}) \\ = \quad & -\mathbf{w}_i' \mathbf{w}_i \{ \frac{\partial \mathbf{S}_i \mathbf{w}_i}{\partial vecut(\mathbf{S}_i)} \frac{1}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i} - \frac{\mathbf{S}_i \mathbf{w}_i}{(\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i)^2} \frac{\partial \mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i}{\partial vecut(\mathbf{S}_i)} \}. \end{aligned}$$

$\partial \mathbf{S}_i \mathbf{w}_i/\partial vecut(\mathbf{S}_i)$ and $\partial \mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i/\partial vecut(\mathbf{S}_i)$ are calculated in Section 5.8.1 and Section 5.8.2.

Block ③ is a $\frac{k(k+1)}{2} \times k$ matrix,

$$\begin{aligned} ③ \quad & \frac{\partial vecut(\mathbf{S}_{i+1})}{\partial \mathbf{w}_i} \quad \text{using Equation (5.4)} \\ = \quad & \frac{\partial}{\partial \mathbf{w}_i} vecut\{(\mathbf{I} - \frac{\mathbf{S}_i \mathbf{w}_i \mathbf{w}_i'}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i}) \mathbf{S}_i (\mathbf{I} - \frac{\mathbf{S}_i \mathbf{w}_i \mathbf{w}_i'}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i})'\} \\ = \quad & \frac{\partial}{\partial \mathbf{w}_i} vecut(\mathbf{S}_i - \frac{\mathbf{S}_i \mathbf{w}_i \mathbf{w}_i' \mathbf{S}_i}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i}) \\ = \quad & -\frac{\partial}{\partial \mathbf{w}_i} vecut(\mathbf{S}_i \mathbf{w}_i \mathbf{w}_i' \mathbf{S}_i) \frac{1}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i} + \frac{vecut(\mathbf{S}_i \mathbf{w}_i \mathbf{w}_i' \mathbf{S}_i)}{(\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i)^2} 2\mathbf{w}_i' \mathbf{S}_i, \\ \text{let} \quad & \mathbf{u}_i = \mathbf{S}_i \mathbf{w}_i, \\ = \quad & -\frac{\partial vecut(\mathbf{u}_i \mathbf{u}_i')}{\partial \mathbf{u}_i} \frac{\partial \mathbf{S}_i \mathbf{w}_i}{\partial \mathbf{w}_i} \frac{1}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i} + \frac{vecut(\mathbf{u}_i \mathbf{u}_i')}{(\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i)^2} 2\mathbf{w}_i' \mathbf{S}_i \\ = \quad & -\frac{\partial vecut(\mathbf{u}_i \mathbf{u}_i')}{\partial \mathbf{u}_i} \frac{\mathbf{S}_i}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i} + \frac{vecut(\mathbf{u}_i \mathbf{u}_i')}{(\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i)^2} 2\mathbf{w}_i' \mathbf{S}_i. \end{aligned}$$

$\partial vecut(\mathbf{u}_i\mathbf{u}_i')/\partial\mathbf{u}_i$ is calculated in Section 5.8.3.

Block ④ is a $\frac{k(k+1)}{2}\times\frac{k(k+1)}{2}$ matrix.

$$\begin{aligned}
④ \quad & \frac{\partial vecut(\mathbf{S}_{i+1})}{\partial vecut(\mathbf{S}_i)} \\
= \quad & \frac{\partial}{\partial vecut(\mathbf{S}_i)} vecut(\mathbf{S}_i - \frac{\mathbf{S}_i\mathbf{w}_i\mathbf{w}_i'\mathbf{S}_i}{\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i}) \\
= \quad & \mathbf{I} - \frac{\partial vecut(\mathbf{u}_i\mathbf{u}_i')}{\partial\mathbf{u}_i}\frac{\partial\mathbf{S}_i\mathbf{w}_i}{\partial vecut(\mathbf{S}_i)}\frac{1}{\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i} + \frac{vecut(\mathbf{u}_i\mathbf{u}_i)}{(\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i)^2}\frac{\partial\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i}{\partial vecut(\mathbf{S}_i)}.
\end{aligned}$$

$\partial vecut(\mathbf{u}_i\mathbf{u}_i')/\partial\mathbf{u}_i$ is shown in Section 5.8.3, $\partial\mathbf{S}_i\mathbf{w}_i/\partial vecut(\mathbf{S}_i)$ is shown in Section 5.8.1, and $\partial\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i/\partial vecut(\mathbf{S}_i)$ is shown in Section 5.8.2.

5.2.3 $\partial\mathbf{w}_l\tilde{\mathbf{R}}/\partial\mathbf{b}$

Let the row vector $\mathbf{d} = \tilde{\mathbf{w}}_l\tilde{\mathbf{R}}$, and its element $d_j = \sum_{i=1}^{a} w_{il}\tilde{r}_{ij}$, where $\tilde{r}_{ij}$ is the element of $\tilde{\mathbf{R}}$ at the i-th row and the j-th column, $j = 1,\ldots,a$.

$$\begin{aligned}
\left(\frac{\partial\mathbf{w}_l\tilde{\mathbf{R}}}{\partial\mathbf{b}}\right)_{\mathbf{b}_0} &= \begin{pmatrix}\frac{\partial d_1}{\partial\mathbf{b}_0} & \frac{\partial d_2}{\partial\mathbf{b}_0} & \cdots & \frac{\partial d_a}{\partial\mathbf{b}_0}\end{pmatrix}. \\
\frac{\partial d_j}{\partial\mathbf{b}_0} &= \sum_{i=1}^{a}\left(\frac{\partial w_{il}}{\partial\mathbf{b}_0}\tilde{r}_{ij} + w_{il}\frac{\partial\tilde{r}_{ij}}{\partial\mathbf{b}_0}\right),
\end{aligned}$$

where Section 5.2.3.1 and Section 5.2.3.2 will show how to calculate $\partial w_{il}/\partial\mathbf{b}_0$, $\tilde{r}_{ij}$ and $\partial\tilde{r}_{ij}/\partial\mathbf{b}_0$.

5.2.3.1 $\partial w_{il}/\partial\mathbf{b}_0$

$\frac{\partial w_{il}}{\partial\mathbf{b}_0}$ can be taken as the l-th row vector from $\frac{\partial\mathbf{w}_i}{\partial\mathbf{b}_{i-1}}\frac{\partial\mathbf{b}_{i-1}}{\partial\mathbf{b}_{i-2}}\cdots\frac{\partial\mathbf{b}_2}{\partial\mathbf{b}_1}$, where $\frac{\partial\mathbf{w}_i}{\partial\mathbf{b}_{i-1}} = \begin{pmatrix}① & ②\end{pmatrix}$ and $\frac{\partial\mathbf{b}_{i-1}}{\partial\mathbf{b}_{i-2}}$ are defined in Section 5.2.2.2.

5.2.3.2 $\tilde{r}_{ij}$ and $\partial\tilde{r}_{ij}/\partial\mathbf{b}_0$

Manne (1987) has given that $\mathbf{R} = \hat{\mathbf{P}}\mathbf{W}$ is an $a\times a$ bidiagonal matrix, whose off-bidiagonal elements r_{ij} are all equal to zero, and bidiagonal elements are r_{ii} and $r_{i(i+1)}$,

$$\begin{cases}
r_{ii} = & 1 & i = 1,\ldots,a \\
r_{i(i+1)=} & \frac{\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_{i+1}}{\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i} & i = 1,\ldots,a-1 \\
r_{ij} = & 0 & \text{otherwise.}
\end{cases}$$

$\tilde{\mathbf{R}} = (\hat{\mathbf{P}}\mathbf{W})^{-1}$ is an $a \times a$ upper triangular matrix, where the upper triangular elements

$$\begin{cases} \tilde{r}_{ij} = 1 & i = j \\ \tilde{r}_{ij} = -\tilde{r}_{i(j-1)} r_{(j-1)j} / r_{jj} & i \neq j. \end{cases}$$

As $\tilde{\mathbf{R}}$ is upper triangular, when $i \geq j$, $\partial \tilde{r}_{ij} / \partial \mathbf{b} = \mathbf{0}$, that is a row vector with $\frac{k(k+3)}{2}$ elements. Because $r_{jj} = 1$, when $i < j$, the derivative of $\tilde{r}_{ij}$ with respect to $\mathbf{b}_0$ can be calculated by an iterative algorithm

$$\begin{aligned} \frac{\partial \tilde{r}_{ij}}{\partial \mathbf{b}_i} &= -\{\frac{\partial \tilde{r}_{i(j-1)}}{\partial \mathbf{b}_i} r_{(j-1)j} + \tilde{r}_{i(j-1)} \frac{\partial r_{(j-1)j}}{\partial \mathbf{b}_i}\}. \\ \frac{\partial \tilde{r}_{ij}}{\partial \mathbf{b}_0} &= \frac{\partial \tilde{r}_{ij}}{\partial \mathbf{b}_i} \frac{\partial \mathbf{b}_i}{\partial \mathbf{b}_{i-1}} \cdots \frac{\partial \mathbf{b}_2}{\partial \mathbf{b}_1}. \end{aligned}$$

$\partial r_{(j-1)j} / \partial \mathbf{b}_i$ can be calculated in the form of $\partial r_{i(i+1)} / \partial \mathbf{b}_i$ as following. As $r_{i(i+1)}$ can be further written as a function of $\mathbf{w}_i$ and $\mathbf{S}_i$,

$$\begin{aligned} r_{i(i+1)} &= \frac{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_{i+1}}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i} = 1 - \frac{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i} \hat{q}_i. \\ \frac{\partial r_{i(i+1)}}{\partial \mathbf{b}_i} &= -\frac{\partial}{\partial \mathbf{b}_i} \frac{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i} \hat{q}_i \\ &= -\frac{\partial \mathbf{w}_i' \mathbf{S}_i \mathbf{S}_i \mathbf{w}_i}{\partial \mathbf{S}_i \mathbf{w}_i} \frac{\partial \mathbf{S}_i \mathbf{w}_i}{\partial \mathbf{b}_i} \frac{\hat{q}_i}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i} - \frac{\mathbf{w}_i' \mathbf{S}_i \mathbf{S}_i \mathbf{w}_i}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i} \frac{\partial \hat{q}_i}{\partial \mathbf{b}_i} + \frac{\mathbf{w}_i' \mathbf{S}_i \mathbf{S}_i \mathbf{w}_i}{(\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i)^2} \hat{q}_i \frac{\partial \mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i}{\partial \mathbf{b}_i} \\ &= -2\mathbf{w}_i' \mathbf{S}_i \begin{pmatrix} \mathbf{S}_i & \frac{\partial \mathbf{S}_i \mathbf{w}_i}{\partial vecut(\mathbf{S}_i)} \end{pmatrix} \frac{\hat{q}_i}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i} - \frac{\mathbf{w}_i' \mathbf{S}_i \mathbf{S}_i \mathbf{w}_i}{\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i} \frac{\partial \hat{q}_i}{\partial \mathbf{b}_i} \\ &\quad + \frac{\mathbf{w}_i' \mathbf{S}_i \mathbf{S}_i \mathbf{w}_i}{(\mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i)^2} \hat{q}_i \begin{pmatrix} 2\mathbf{w}_i' \mathbf{S}_i & \frac{\partial \mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i}{\partial vecut(\mathbf{S}_i)} \end{pmatrix}, \end{aligned}$$

where $\frac{\partial \mathbf{S}_i \mathbf{w}_i}{\partial vecut(\mathbf{S}_i)}$ and $\frac{\partial \mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i}{\partial vecut(\mathbf{S}_i)}$ are shown Section 5.8.1 and Section 5.8.2.

5.3 New Linearisation Method Bootstrapping Estimate

The key idea of the new linearisation approximation to prediction uncertainty is to find the variance of the estimated regression coefficients Var $(\hat{\boldsymbol{\beta}})$. The mathematical derivation as shown in the previous sections is a straightforward approach to obtain the estimated regression coefficient variance, but it is computationally expensive. Bootstrapping can be used as an alternative to avoid

complicated calculations. For the B-th bootstrapping sample, $\mathbf{b}_{0,B}$ is drawn from a Wishart distribution given by the asymptotic result of $\mathrm{Var}(\mathbf{b}_0)$ in Section 5.2.1, where $B = 1, \ldots, M$. The regression coefficient estimate $\hat{\boldsymbol{\beta}}_B$ is calculated from a reformatted univariate partial least squares algorithm each time, and then $\mathrm{Var}(\hat{\boldsymbol{\beta}}) = \frac{1}{M-1}\sum_{B=1}^{M}(\hat{\boldsymbol{\beta}}_B - \bar{\boldsymbol{\beta}})(\hat{\boldsymbol{\beta}}_B - \bar{\boldsymbol{\beta}})'$ can be plugged into the prediction mean squared error formula Equation (5.2), where $\bar{\boldsymbol{\beta}} = \frac{1}{M}\sum_{B=1}^{M}\hat{\boldsymbol{\beta}}_B$. Partial least squares regression orthogonal scores reformatted algorithm is a function of the sum of squares $\mathbf{b}_0$ (See Equations (5.3) and (5.4)). It is not directly connected to the centred data $(\mathbf{X}_c, \mathbf{y}_c)$.

Algorithm 5.1. Bootstrapping Orthogonal Scores Univariate Partial Least Squares Algorithm

For $i = 1, \ldots, a$, $\mathbf{w}_1$ consists of the 1-st to k-th elements of $\mathbf{b}_0$ corresponding to $\mathbf{w}_1 = \mathbf{X}_c'\mathbf{y}_c$, and $\mathbf{S}_1$ is a square matrix built by the $(k+1)$-th to $\{k(k+3)/2\}$-th elements of $\mathbf{b}_0$; When $i \geq 2$, $\mathbf{w}_i = \mathbf{A}_{i-1}\mathbf{w}_{i-1}$, and $\mathbf{S}_i = \mathbf{A}_{i-1}\mathbf{S}_{i-1}\mathbf{A}_{i-1}$.

- $\mathbf{A}_i = \mathbf{I} - \mathbf{S}_i\mathbf{w}_i\mathbf{w}_i'/\mathbf{w}_i'\mathbf{S}_i\mathbf{w}_i$.
- $\mathbf{v}_i = \mathbf{w}_i/\sqrt{\mathbf{w}_i'\mathbf{w}_i}$.
- $\hat{\mathbf{p}}_i = \mathbf{S}_i\mathbf{v}_i/\mathbf{v}_i'\mathbf{S}_i\mathbf{v}_i$.
- $\hat{q}_i = \mathbf{w}_i'\mathbf{v}_i/\mathbf{v}_i'\mathbf{S}_i\mathbf{v}_i$.

That $\mathbf{v}_i'\mathbf{v}_i = 1$ brings more stability for the orthogonal scores algorithm. For the b-th bootstrapping sample, $\hat{\boldsymbol{\beta}}^b = \mathbf{V}(\hat{\mathbf{P}}'\mathbf{V})^{-1}\hat{\mathbf{q}}$, where $b = 1, \cdots, M$. As shown in Equation (4.9), $\mathrm{Var}(\hat{\boldsymbol{\beta}}) = \frac{1}{M-1}\sum_{b=1}^{M}(\hat{\boldsymbol{\beta}}^b - \bar{\boldsymbol{\beta}})(\hat{\boldsymbol{\beta}}^b - \bar{\boldsymbol{\beta}})'$.

5.4 Univariate Partial Least Squares Regression Prediction Mean Squared Error Summary

This section will give a list of prediction uncertainty quantification methods used in the simulation study and the real data analysis. As the estimated regression error

variance $\hat{\sigma}_\epsilon^2$ is important in these prediction mean squared error formulae, we will talk about it firstly. In Section 4.2.2 we introduces the ordinary least squares type prediction mean squared error with two estimated regression error variances. The direct estimate of regression error variance from the calibration set underestimates the true value, so the adjusted estimate from a tuning set $\{\dot{\mathbf{X}}_t, \dot{\mathbf{y}}_t\}$ with the sample size n_t, can be useful. The advantage of the adjustment is to ensure the relationship between prediction mean squared error and leverage to be correct on average, and it also takes the bias into account. To use the estimated regression error variance from the tuning set in the bootstrapping by residuals method, Denham (1997)'s method, the new linearisation method, and its bootstrapping version, would also give better results, which compensates the bias.

Let **SPE** denote the squared prediction error. Let **OLS** denote the ordinary least squares type prediction mean squared error. Let Bootstrapping denote the prediction mean squared error calculated by the bootstrapping by residuals. Let **Lin1** denote the prediction mean squared error given by Denham (1997). Let **Lin2** denote the new prediction mean squared error.

- **SPE** - squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$.

- **OLS** - the ordinary least squares type prediction mean squared error as shown in Equation (4.3), and its estimated regression error variance from the tuning set as shown in Equation (4.19).

$$\begin{aligned} \mathrm{E}\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} &= \sigma_{\epsilon_o}^2\{\frac{1}{n} + \underbrace{\mathbf{t}_p(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_p'}_{h} + 1\}. \qquad (5.8) \\ \hat{\sigma}_{\epsilon_o}^2 &= \frac{\frac{1}{n_t}\sum_{j=1}^{n_t}(\dot{y}_{t_j} - \hat{\dot{y}}_{t_j})^2}{1 + \frac{1}{n_t} + \frac{1}{n_t}\sum_{j=1}^{n_t} h_{t_j}}. \\ \text{where } h &= \mathbf{t}_p(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_p', \qquad (5.9) \\ \text{and } h_t &= \mathbf{t}_t(\mathbf{T}'\mathbf{T})^{-1}\mathbf{t}_t'. \end{aligned}$$

  The scores of the tuning set $\mathbf{t}_t = \mathbf{x}_t\mathbf{W}(\hat{\mathbf{P}}'\mathbf{W})^{-1}$.

- **Bootstrapping** - the bootstrapping by residuals prediction mean squared

error as shown in Equation (4.10)

$$\begin{aligned} \mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} &= \sigma^2_{\epsilon_b}(1+\frac{1}{n}) + \underbrace{\mathbf{x}_p \mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B)\mathbf{x}'_p}_{\mathrm{hb}}. && (5.10)\\ \hat{\sigma}^2_{\epsilon_b} &= \frac{\frac{1}{n_t}\sum_{j=1}^{n_t}\{(\dot{y}_{t_j} - \hat{\dot{y}}_{t_j})^2 - \mathrm{hb}_{t_j}\}}{1+\frac{1}{n_t}}, && (5.11)\\ \text{where hb} &= \mathbf{x}_p \mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B)\mathbf{x}'_p, && (5.12)\\ \text{and hb}_t &= \mathbf{x}_t \mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B)\mathbf{x}'_t. \end{aligned}$$

$\mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B)$ is the regression variance estimate calculated from the calibration set using bootstrapping by residuals (See Section 4.2.4). Each bootstrapping dataset calculates $\hat{\boldsymbol{\beta}}^b$ from Algorithm 4.2, and then $\mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B) = \frac{1}{M-1}\sum_{B=1}^{M}(\hat{\boldsymbol{\beta}}^b - \bar{\boldsymbol{\beta}})(\hat{\boldsymbol{\beta}}^b - \bar{\boldsymbol{\beta}})'$ as shown in Equation (4.9).

- **Lin1** - the classical local linearisation prediction mean squared error proposed by Denham (1997) as shown in Equation (4.7),

$$\begin{aligned} \mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} &= \sigma^2_{\epsilon_{l1}}(1+\frac{1}{n}+\underbrace{\mathbf{x}_p\mathbf{J}\mathbf{J}'\mathbf{x}'_p}_{\mathrm{Hden}}). && (5.13)\\ \hat{\sigma}^2_{\epsilon_{l1}} &= \frac{\frac{1}{n_t}\sum_{j=1}^{n_t}(\dot{y}_{t_j} - \hat{\dot{y}}_{t_j})^2}{1+\frac{1}{n_t}+\frac{1}{n_t}\sum_{j=1}^{n_t}\mathrm{Hden}_{t_j}}, && (5.14)\\ \text{where Hden} &= \mathbf{x}_p\mathbf{J}\mathbf{J}'\mathbf{x}'_p, && (5.15)\\ \text{and Hden}_t &= \mathbf{x}_t\mathbf{J}\mathbf{J}'\mathbf{x}'_t. \end{aligned}$$

.

- **Lin2** - the new local linearisation method predication variance as shown in Equation (5.2),

$$\begin{aligned} \mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} &= \sigma^2_{\epsilon_{l2}}(1+\frac{1}{n}) + \underbrace{\mathbf{x}_p\big(\frac{\partial\hat{\boldsymbol{\beta}}}{\partial\mathbf{b}}\big)_{\mathbf{b}_0}\mathrm{Var}\,(\mathbf{b}_0)\big(\frac{\partial\hat{\boldsymbol{\beta}}}{\partial\mathbf{b}}\big)'_{\mathbf{b}_0}\mathbf{x}'_p}_{\mathrm{H}}. && (5.16)\\ \hat{\sigma}^2_{\epsilon_{l2}} &= \frac{\frac{1}{n_t}\sum_{j=1}^{n_t}\{(\dot{y}_{t_j} - \hat{\dot{y}}_{t_j})^2 - \mathrm{H}_{t_j}\}}{1+\frac{1}{n_t}}. && (5.17)\\ \text{where H} &= \mathbf{x}_p\big(\frac{\partial\hat{\boldsymbol{\beta}}}{\partial\mathbf{b}}\big)_{\mathbf{b}_0}\mathrm{Var}\,(\mathbf{b}_0)\big(\frac{\partial\hat{\boldsymbol{\beta}}}{\partial\mathbf{b}}\big)'_{\mathbf{b}_0}\mathbf{x}'_p, && (5.18)\\ \text{and H}_t &= \mathbf{x}_t\big(\frac{\partial\hat{\boldsymbol{\beta}}}{\partial\mathbf{b}}\big)_{\mathbf{b}_0}\mathrm{Var}\,(\mathbf{b}_0)\big(\frac{\partial\hat{\boldsymbol{\beta}}}{\partial\mathbf{b}}\big)'_{\mathbf{b}_0}\mathbf{x}'_t. \end{aligned}$$

.

- **Lin2b** - the new local linearisation method prediction variance calculated from the bootstrapping method as shown in Equation (4.10),

$$\begin{aligned} \mathrm{E}\left\{(\dot{y}_p - \hat{\dot{y}}_p)^2\right\} &= \sigma^2_{\epsilon_{l2b}}(1+\frac{1}{n}) + \underbrace{\mathbf{x}_p \mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B)\mathbf{x}'_p}_{\mathrm{Hb}}. \qquad (5.19) \\ \hat{\sigma}^2_{\epsilon_{l2b}} &= \frac{\frac{1}{n_t}\sum_{j=1}^{n_t}\{(\dot{y}_{t_j} - \hat{\dot{y}}_{t_j})^2 - \mathrm{Hb}_{t_j}\}}{1+\frac{1}{n_t}}, \qquad (5.20) \\ \text{where Hb} &= \mathbf{x}_p \mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B)\mathbf{x}'_p, \qquad (5.21) \\ \text{and Hb}_t &= \mathbf{x}_t \mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B)\mathbf{x}'_t \end{aligned}$$

  and $\mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B)$ is the bootstrapping regression coefficient variance estimate of the calibration set. According to Section 5.3, $\hat{\boldsymbol{\beta}}^b$ is calculated by Algorithm 5.1, and then $\mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B) = \frac{1}{M-1}\sum_{B=1}^{M}(\hat{\boldsymbol{\beta}}^b - \bar{\boldsymbol{\beta}})(\hat{\boldsymbol{\beta}}^b - \bar{\boldsymbol{\beta}})'$ shown in Equation (4.9). Although Bootstrapping and Lin2b share the form of the ordinary least squares type prediction mean squared error, the two variance estimates of regression coefficients $\mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B)$ are calculated from different bootstrapping procedures, so hb (Equation (5.12)) and Hb (Equation (5.21)) are not the same.

We will compare these methods in the simulation study and the real data analysis. In univariate partial least squares regression, the OLS prediction mean squared error is linear with the leverage h, but in the linearisation methods the leverage is not directly connected to the prediction mean squared error. For the linearisation methods, the predication variances are linear with the distance measures $\mathrm{Hden} = \mathbf{x}_p\mathbf{J}\mathbf{J}'\mathbf{x}'_p$, $\mathrm{H} = \mathbf{x}_p\big(\frac{\partial\hat{\boldsymbol{\beta}}}{\partial\mathbf{b}}\big)_{\mathbf{b}_0}\mathrm{Var}\,(\mathbf{b}_0)\big(\frac{\partial\hat{\boldsymbol{\beta}}}{\partial\mathbf{b}}\big)'_{\mathbf{b}_0}\mathbf{x}'_p$, and $\mathrm{Hb} = \mathbf{x}_p\mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B)\mathbf{x}'_p$. Bootstrapping prediction mean squared error is linear with $\mathrm{hb} = \mathbf{x}_p\mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B)\mathbf{x}'_p$. To study these linear relationships, average squared prediction error against average distance measure plot will be used. On the other hand, in order to observe how h, hb, Hden, H and Hb are associated with each other, we will choose a metric, and decompose other distance measures into the direction of this metric. For instance, in the average squared prediction error against average leverage plot,

the OLS prediction mean squared error would give the linear relationship shown in Equation (4.3); Bootstrapping, Lin1, Lin2 and Lin2b would demonstrate how these prediction variances change with hb, Hden, H, and Hb in the direction of h, respectively. Why do we need them to be presented separately?

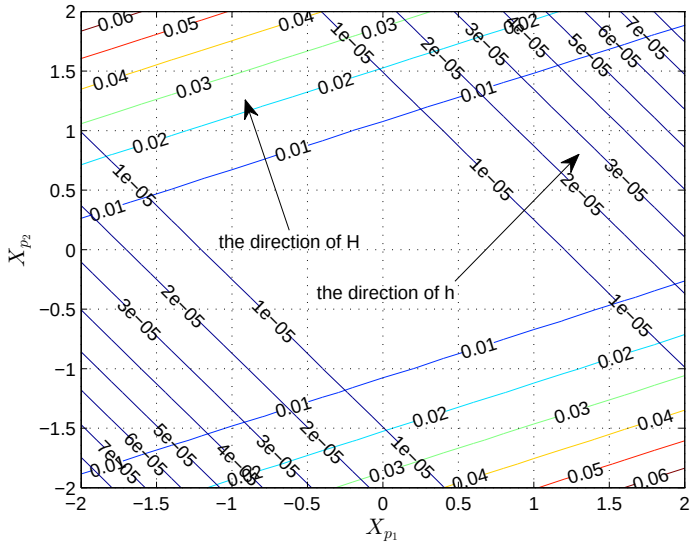


Figure 5.1: PLS: a Contour Plot to Show the Directions of h and H. $k = 2$, $a = 1$, $\sigma^2_{c_1} = \sigma^2_{c_2} = 1$, $\sigma^2_\epsilon = 0.0025$, $\beta_0 = \beta_1 = \beta_2 = 1$.

Before answering the question, let us have a look at the contour plot of h and H, Figure 5.1, where $k = 2$, $a = 1$, $\sigma^2_{c_1} = \sigma^2_{c_2} = 1$, $\sigma^2_\epsilon = 0.0025$, $\beta_0 = \beta_1 = \beta_2 = 1$. The x-axis and y-axis represent the predictors $\dot{X}_{p_1}$ and $\dot{X}_{p_2}$ in the range of $[-2, 2]$. The sideways tilted contours in light colours are lines joining the points of equal H values. The vertically inclined contours in deep colours are presented for h. The directions of h and H are given by two black arrows, perpendicular to the contours. It can be seen that there is a big angle between the direction of h and the direction of H, which tells if prediction mean squared error goes up with an increasing leverage, it does not mean prediction mean squared error would rise when H value increases. Both h and H are distance measures, but they are quite different, thus it is necessary to study their relationship with prediction mean squared error individually. This also answers the question why we need to plot squared prediction error against h, hb, Hden, H, and Hb one by one, to study

prediction uncertainty.

Furthermore, getting to know the different directions of these metrics helps understand more about these formulae.

The OLS prediction mean squared error is linear with the leverage. $\sigma^2_{\epsilon_o}$ plays the role of a projector that converts the useful part of the leverage into prediction mean squared error. $\sigma^2_{\epsilon_o}$ gives the unit that how much leverage contributes to prediction mean squared error. The estimated regression error variance from the tuning set $\hat{\sigma}^2_{\epsilon_o}$ is useful because it not only completes the projection task but also compensates for the bias at the same time.

In the new linearisation prediction mean squared error formula, the quantification of H contains both the variation about the regression and the variation in the explanatory and response variables. If H is used as the metric, without considering the estimate of $\sigma^2_{\epsilon_{l2}}$, the prediction mean squared error is proportional to H with the slope of 1. Intuitively, it tells at most all H can be used in the measurement of prediction mean squared error, ideally, if the direction of H is the same as that of the prediction mean squared error. If not, how much H can be used in the measurement of prediction mean squared error? Unlike the OLS prediction mean squared error where $\sigma^2_{\epsilon_o}$ acts as the projector, the new linearisation prediction mean squared error formula lacks such an adjusted factor. Hence, the estimated regression error variance $\hat{\sigma}^2_{\epsilon_{l2}}$ become essential in the linearisation prediction mean squared error formula, because it contains an average H_t obtained from the tuning set, functioning as the adjusted factor.

5.5 Univariate Partial Least Squares Regression Simulation Study

The design of uniformly distributed leverage gives a special relationship between the bias and the leverage, so the use of multivariate normally distributed predictors is more appropriate for both of principal components regression and partial least squares regression. Thus, we generate multivariate normally distributed predictors

directly in the univariate partial least squares regression simulation. The trilinear model for a single response variable partial least squares regression can be written as

$$\dot{\mathbf{y}}_c = \beta_0 + \dot{\mathbf{X}}_c\boldsymbol{\beta} + \boldsymbol{\epsilon}_c,$$
$$\dot{\mathbf{y}}_t = \beta_0 + \dot{\mathbf{X}}_t\boldsymbol{\beta} + \boldsymbol{\epsilon}_t,$$
$$\dot{\mathbf{y}}_p = \beta_0 + \dot{\mathbf{X}}_p\boldsymbol{\beta} + \boldsymbol{\epsilon}_p,$$

where $\dot{\mathbf{y}}_c$, $\dot{\mathbf{y}}_t$ and $\dot{\mathbf{y}}_p$ are calibration, tuning and prediction response variables. $\dot{\mathbf{X}}_c$ $(n \times k)$, $\dot{\mathbf{X}}_t$ $(n_t \times k)$ and $\dot{\mathbf{X}}_p$ $(n_p \times k)$ are calibration, tuning and prediction explanatory variables matrices. β_0 and $\boldsymbol{\beta}$ $(k \times 1)$ are regression coefficients. $\boldsymbol{\epsilon}_c$ and $\boldsymbol{\epsilon}_p$ are error terms of the calibration set and the prediction set. Assume the tuning set is the same as the calibration set because the samples drawn from the same distribution and with the same regression structure would be of interest. Let i denote the number of replicates. In total, there are N replicates. Let j denote the number of observations: in the calibration set $j = 1, \ldots, n$; in the tuning set $j = 1, \ldots, n_t$; in the prediction set $j = 1, \ldots, n_p$. Each calibration set simulates its own explanatory variables, referring to Case (3) in the ordinary least squares regression simulation study (See Simulation 2.1).

1. The simulation of calibration sets and tuning sets

   Explanatory variables $\dot{\mathbf{X}}_c$ are independent and identical normally distributed with mean $\mathbf{0}$ and variances ($\sigma^2_{c_1}$ $\sigma^2_{c_2}$ $\cdots$ $\sigma^2_{c_k}$). The noise $\boldsymbol{\epsilon}$ is also independent and identical normally distributed with with mean 0 and variance σ^2_ϵ. Each observation in the calibration set can be calculated as $\dot{y}_c = \beta_0 + \dot{\mathbf{x}}_c\boldsymbol{\beta} + \epsilon_c$. The tuning sets are generated in the same way as the calibration sets, hence $\dot{y}_t = \beta_0 + \dot{\mathbf{x}}_t\boldsymbol{\beta} + \epsilon_t$.

2. The simulation of prediction sets

   We assume the predictor $\dot{\mathbf{X}}_p$ have the same distribution as $\dot{\mathbf{X}}_c$. The noise $\boldsymbol{\epsilon}_p$ is independent and identical normally distributed with with mean 0 and variance σ^2_ϵ. Each observation in the prediction set can be calculated as $\dot{y}_p = \beta_0 + \dot{\mathbf{x}}_p\boldsymbol{\beta} + \epsilon_p$.

3. The choice of the number of factors

   Leave-one-out cross-validation is employed to choose the number of factors, where RMSECV defined in Section 4.2.1 is calculated under a series of a number of factors. The minimum and the maximum of RMSECV are denoted as RMSECVmin and RMSECVmax. The range RMSECVr = RMSECVmax − RMSECVmin. The number of factors a is chosen to be the smallest integer whose RMSECV is equal to or bigger than RMSECVmin $+ 0.1 \times$ RMSECVr. This sets up the minimum floating up ten percent of the range as the threshold.

4. The calibration and the prediction

   After the number of factors is chosen to be a, the partial least squares algorithm (See Section 4.1) gives the score matrix $\mathbf{T}$, the weight matrix $\mathbf{W}$, the x-loading matrix $\hat{\mathbf{P}}$, and the y-loading vector $\hat{\mathbf{q}}$. Take the orthogonal scores algorithm (Algorithm 4.2) for example, the score of the predictor $\mathbf{x}_p$ can be calculated as $\mathbf{t}_p = \mathbf{x}_p\mathbf{W}(\hat{\mathbf{P}}'\mathbf{W})^{-1}$, so the predicted value $\hat{\dot{y}}_p = \bar{\dot{y}} + \mathbf{t}_p\hat{\mathbf{q}}'$. Likewise, the fitted value of the tuning set can be calculated too.

We will study the statistical behavior of univariate partial least squares regression prediction variances presented by the ordinary least squares type expression (**OLS**), the bootstrapping by residuals (**Bootstrapping**), Denham's linearisation method (**Lin1**), the new local linearisation method (**Lin2**) and its bootstrapping version (**Lin2b**).

As these prediction mean squared error formulae describe average behaviour, and under the normality assumption, the leverage has a Chi-square distribution with a degrees of freedom. Chi-square binning method defined in Simulation 2.1 of Section 2.2 will assist in presenting the average results. Section 5.4 has shown h, hb, Hden, H, and Hb are different distance measures. In common, they are all quadratic functions of centred predictors $\mathbf{x}_p$. The distributions of hb, Hden, H and Hb can be regarded as transformations from the Chi-square distribution, although the mathematical forms of the four distance measures are complicated.

Hence, the Chi-square binning method can be approximately used for hb, Hden, H, and Hb as well. We will use average squared prediction error against average leverage plot, average squared prediction error against average hb plot, average squared prediction error against average Hden plot, average squared prediction error against average H plot and average squared prediction error against average Hb plot to compare these prediction variances summarised in Section 5.4.

5.5.1 Partial Least Squares Regression Simulation with Noise Free Prediction Samples

To present the simplest relationship, we study the noise free simulation firstly, where in the prediction set the error term $\boldsymbol{\epsilon}_p$ is set to be 0. To keep the actual squared prediction error with the level of the calibration set, we add 0.25 into the squared prediction error in the prediction set where $\boldsymbol{\epsilon}_p = \mathbf{0}$. The sample sizes are set as $n = 200$, $n_t = 200$, and $n_p = 200$.

Simulation 5.1. $k = a = 1$, $\sigma_c^2 = 1$, $\beta_0 = \beta_1 = 1$, , $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$, $N = 10000$

When $k = a = 1$, partial least squares regression is equivalent to the simple ordinary least squares regression. $\mathrm{Var}\,(\hat{\beta})$ in the ordinary least squares type expression equals to

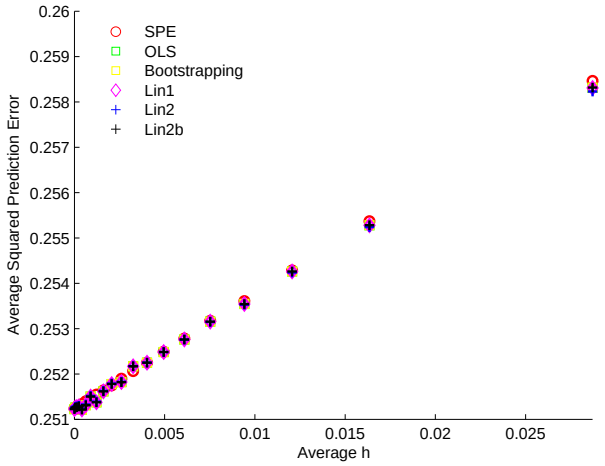
$$\mathrm{Var}\,(\hat{\beta}) = \hat{\sigma}_{\epsilon_1}^2 / S_{xx}, \tag{5.22}$$

where $\hat{\sigma}_{\epsilon_1}^2 = \frac{1}{n-1}(s_{yy} - \frac{s_{xy}^2}{S_{xx}})$. $\mathrm{Var}\,(\hat{\beta})$ in Denham's method can be calculated as $\hat{\sigma}_{\epsilon_1}^2 / S_{xx}$ too. The new linearisation method gives

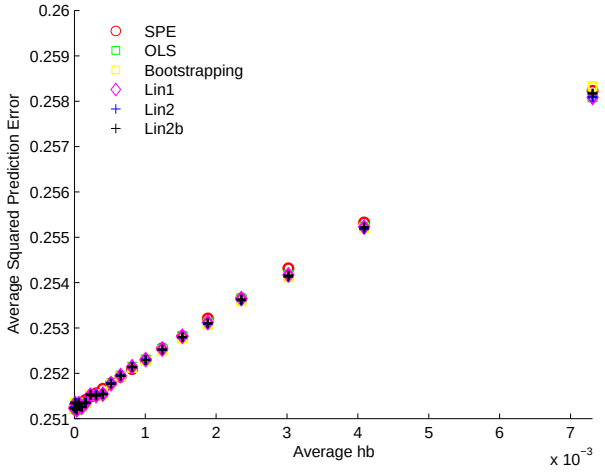
$$\mathrm{Var}\,(\hat{\beta}) = \frac{n}{(n-1)^2}(\frac{s_{yy}}{S_{xx}} - \frac{s_{xy}^2}{S_{xx}^2}) = \hat{\sigma}_{\epsilon_2}^2 / S_{xx}. \tag{5.23}$$

where $\hat{\sigma}_{\epsilon_2}^2 = \frac{n}{(n-1)^2}(s_{yy} - \frac{s_{xy}^2}{S_{xx}})$. When n is large, $\hat{\sigma}_{\epsilon_1}^2 \approx \hat{\sigma}_{\epsilon_2}^2$. Hence, Equation (5.22) and Equation (5.23) are approximate in the same.

In Figure 5.2, the red circle line (SPE) presents for the relationship between average squared prediction error and average distance measure of interest. The green square point line (OLS) denotes the ordinary least squares type prediction

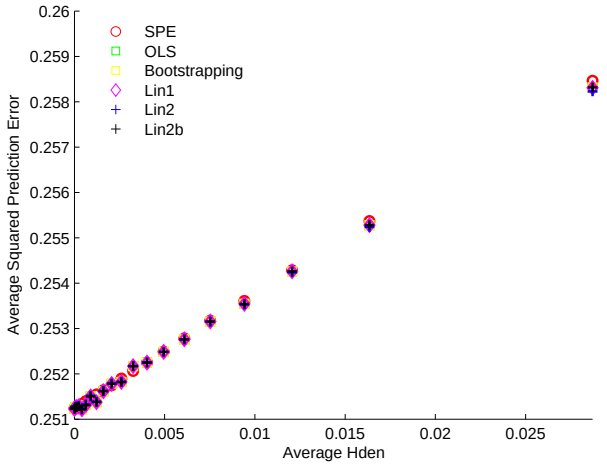


(a) average leverage h, Equation (5.9)

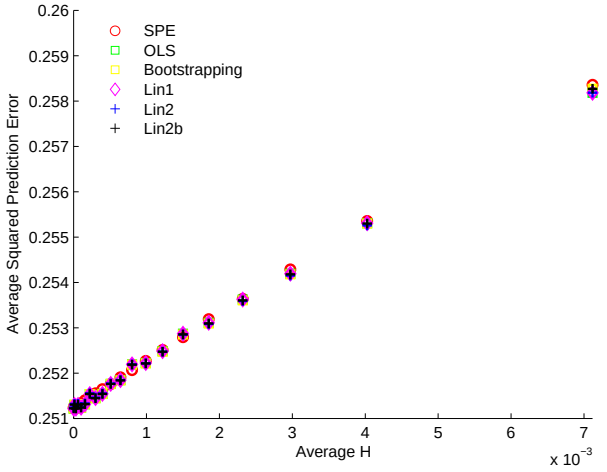


(b) average hb, Equation (5.12)

Figure 5.2: PLS Average Squared Prediction Error versus Average Distance Measure. $k = a = 1$ $\mathrm{Var}\,(\dot{X}_c) = 1$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$. OLS: the ordinary least squares type prediction mean squared error, Equation (5.8). Bootstrapping: the bootstrapping-by-residual prediction mean squared error, Equation (5.10). Lin1: Denham's prediction mean squared error, Equation (5.13). Lin2: the new linearisation prediction mean squared error, Equation (5.16). Lin2b: the new linearisation prediction mean squared error embedded with bootstrapping, Equation (5.19).

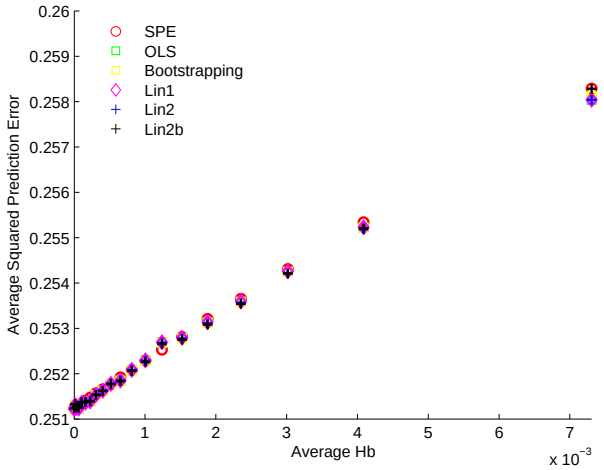


(c) average Hden, Equation (5.15)



(d) average H, Equation (5.18)

Figure 5.2: PLS Average Squared Prediction Error versus Average Distance Measure. $k = a = 1$ $\mathrm{Var}\,(\dot{X}_c) = 1$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).



(e) average Hb, Equation (5.21)

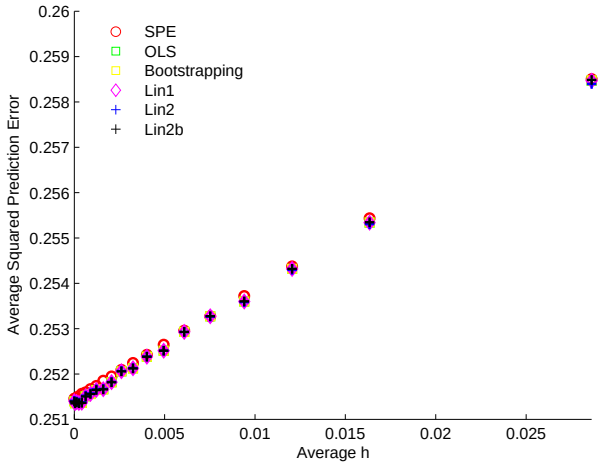
Figure 5.2: PLS Average Squared Prediction Error versus Average Distance Measure. $k = a = 1$ $\mathrm{Var}\,(\dot{X}_c) = 1$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).

mean squared error. The yellow square point line (Bootstrapping) gives the prediction mean squared error calculated from the bootstrapping by residuals. The magenta diamond point line (Lin1) stands for the prediction mean squared error of the classical linearisation method proposed by Denham (1997). The blue plus point line (Lin2) displays the prediction mean squared error given by the new linearisation method. The black plus point line (Lin2b) gives the prediction mean squared error calculated from the new linearsiation method bootstrapping version. In Figure 5.2(a) - (e), SPE, OLS, Bootstrapping, Lin1, Lin2 and Lin2b overlap, which is consistent with the theoretical results.

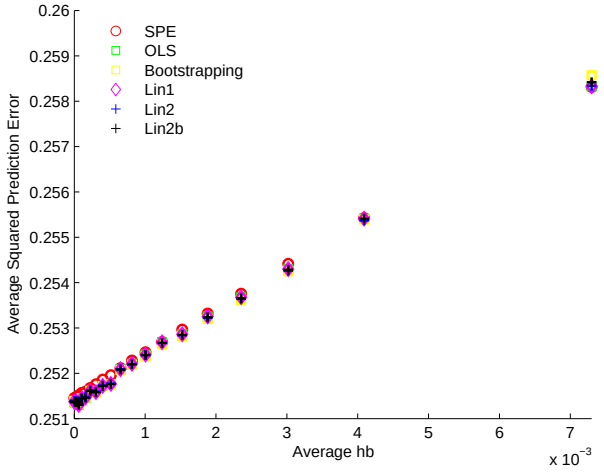
Simulation 5.2. $k = 2$, $a = 1$, $\sigma_{c_1}^2 = 25$, $\sigma_{c_2}^2 = 1$, $\beta_0 = \beta_1 = 1$, $\beta_2 = 0$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$, $N = 10000$.

Figure 5.3 gives similar results to Simulation 5.1, because $\dot{X}_{c_1}$ has a much larger variance than $\dot{X}_{c_2}$, $\beta_1 = 1$, and $\beta_2 = 0$, making it almost the case of $k = a = 1$. The introduction of an extra explanatory variable $\dot{X}_{c_2}$ makes them a bit more noisy.

Figure 5.4 gives how the estimated regression coefficients change with the newly constructed $\mathbf{b}$ when $k = 2$, $a = 1$. $\hat{\beta}_1$ shifting around 1 and $\hat{\beta}_2$ moving around 0

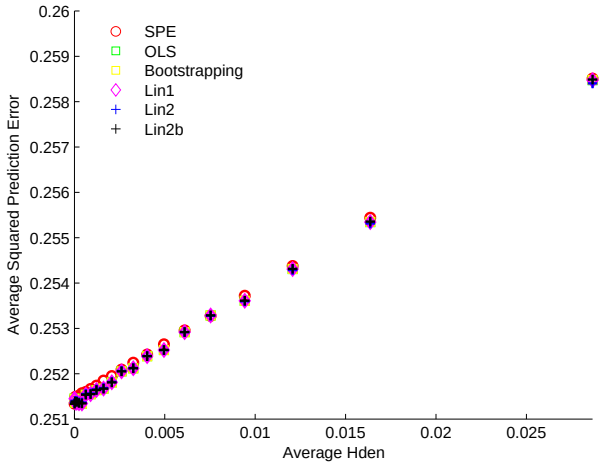


(a) average leverage h, Equation (5.9)

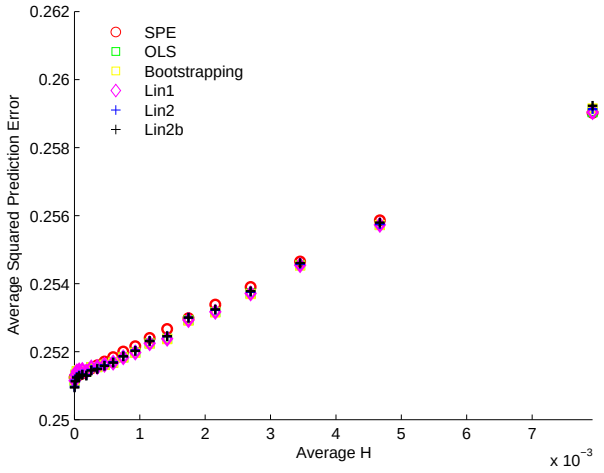


(b) average hb, Equation (5.12)

Figure 5.3: PLS Average Squared Prediction Error versus Average Distance Measure. $k = 2$, $a = 1$, $\mathrm{Var}\,(\dot{\mathbf{X}}_{c_1}) = 25$, $\mathrm{Var}\,(\dot{\mathbf{X}}_{c_2}) = 1$, $\beta_0 = \beta_1 = 1$, $\beta_2 = 0$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$. OLS: the ordinary least squares type prediction mean squared error, Equation (5.8). Bootstrapping: the bootstrapping-by-residual prediction mean squared error, Equation (5.10). Lin1: Denham's prediction mean squared error, Equation (5.13). Lin2: the new linearisation prediction mean squared error, Equation (5.16). Lin2b: the new linearisation prediction mean squared error embedded with bootstrapping, Equation (5.19).

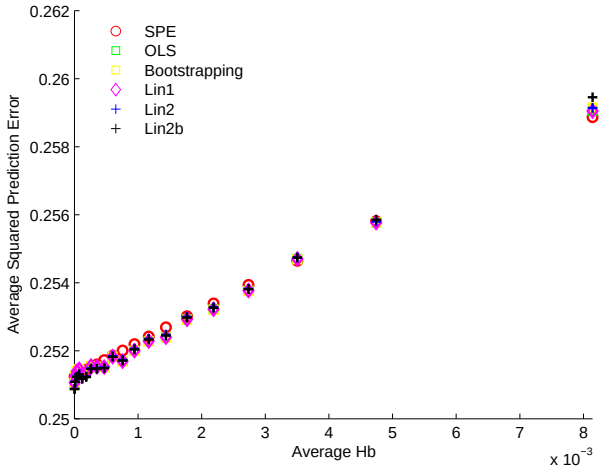


(c) average Hden, Equation (5.15)



(d) average H, Equation (5.18)

Figure 5.3: PLS Average Squared Prediction Error versus Average Distance Measure. $k = 2$, $a = 1$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_1}\right) = 25$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_2}\right) = 1$, $\beta_0 = \beta_1 = 1$, $\beta_2 = 0$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).



(e) average Hb, Equation (5.21)

Figure 5.3: PLS Average Squared Prediction Error versus Average Distance Measure. $k = 2$, $a = 1$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_1}\right) = 25$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_2}\right) = 1$, $\beta_0 = \beta_1 = 1$, $\beta_2 = 0$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).

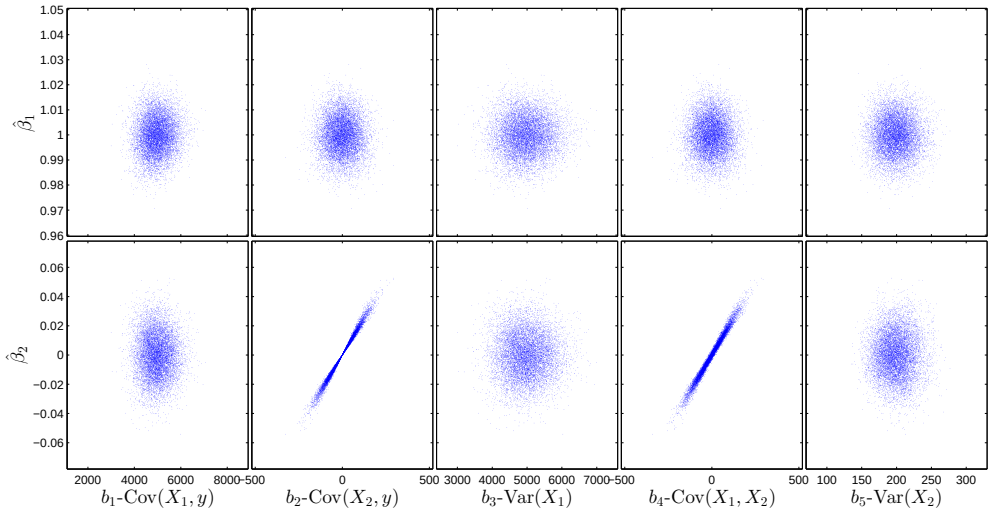


Figure 5.4: PLS $\hat{\boldsymbol{\beta}}$ against $\mathbf{b}$ when $k = 2$, $a = 1$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_1}\right) = 25$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_2}\right) = 1$, $\beta_0 = \beta_1 = 1$, $\beta_2 = 0$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$.

centro-symmetrically tells that the linearisation method works well.

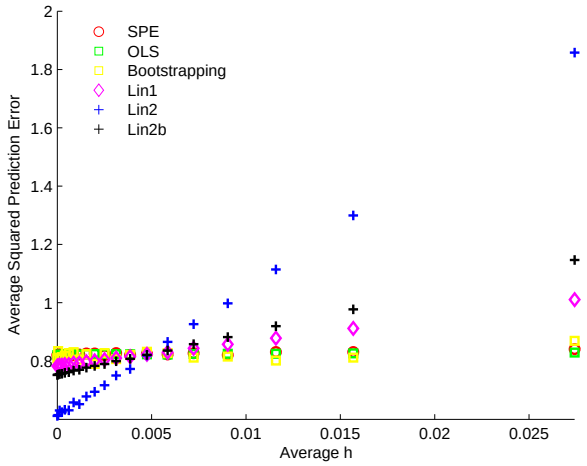
Simulation 5.3. $k = 2$, $a = 1$, $\sigma^2_{c_1} = 25$, $\sigma^2_{c_2} = 1$, $\beta_1 = 0$, $\beta_0 = \beta_2 = 1$, $\sigma^2_\epsilon = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$, $N = 10000$

Under the worst circumstances, how explanatory variables interact with the response variable is unclear. Partial least squares regression would be completely confused, and cannot generate appropriate components. In Figure 5.5, $\dot{X}_{c_1}$ has the largest variance but has no contribution to the regression, whereas $\dot{X}_{c_2}$ with a smaller variance is actually the only variable linked to the response variable. As $\dot{X}_{c_1}$ has a bigger variance, its covariance with the response variable will mess up partial least squares regression to help $\dot{X}_{c_1}$ gain more weights in the new factor than that is actually taken up by $\dot{X}_{c_1}$. Figure 5.5(a) shows OLS is not affected. Figure 5.5(b) tells Bootstrapping does not work. In Figure 5.5(c) Lin1 works all right, except that Lin1 a bit underestimates squared prediction errors before the average Hden, and it gives slightly larger estimates after the average Hden. In Figure 5.5(d) Lin2 increases dramatically showing the new linearisation method over-estimates prediction uncertainty. Figure 5.5(e) tells Lin2b also fails.

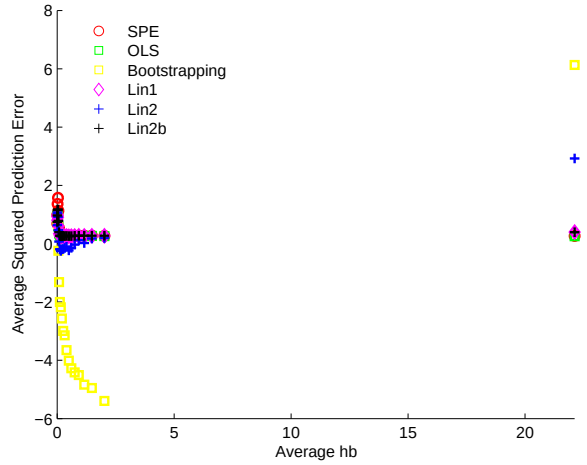
The bi-mode trend shown in Figure 5.6 confirms the disfunction of partial least squares regression. $\hat{\beta}_1$ shifts between -0.1274 and 0.1326. $\hat{\beta}_2$ slips between 0.0354 and 1.1022. Taking $\hat{\beta}_1$ and $\hat{\beta}_2$ against b_1 for example, $\hat{\beta}_1$ against b_1 has a 'Z' shape, and $\hat{\beta}_2$ against b_1 has a 'Λ' shape. When $\hat{\beta}_1$ takes the value on the two legs of 'Z', $\hat{\beta}_2$ has the values on the two tails of 'Λ', that are close to 0. The fat 'Z' legs show that the majority of $\hat{\beta}_1$ are around -0.1 and 0.1, correspondingly $\hat{\beta}_2$ are less than 1. When $\hat{\beta}_1$ is plotted against b_2, b_3 and b_5, trapezoids with fat legs are sketched, which also indicates the partial least squares regression has been messed up.

Simulation 5.4. $k = 3$, $a = 2$, $\sigma^2_{c_1} = \sigma^2_{c_2} = 25$, $\sigma^2_{c_3} = 1$, $\beta_0 = \beta_1 = \beta_2 = 1$, $\beta_3 = 0$, $\sigma^2_\epsilon = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$, $N = 10000$

When explanatory variables have equal or close variances, the new linearisation method may fail to give a good linear approximation to the variance of the estimated regression coefficients. Figure 5.7 is the results from the simulation where $\dot{X}_{c_1}$ and $\dot{X}_{c_2}$ are assumed to have the same large variance.

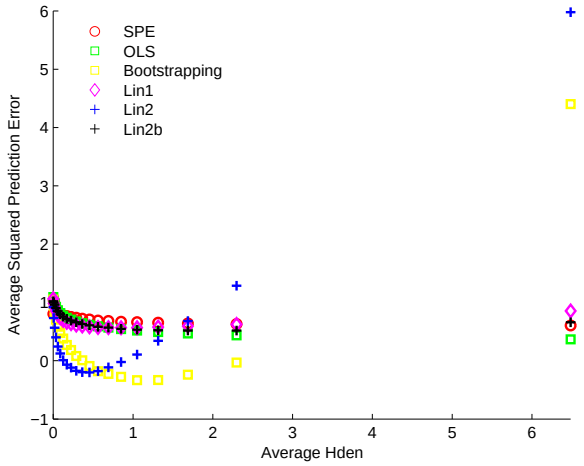


(a) average leverage h, Equation (5.9)

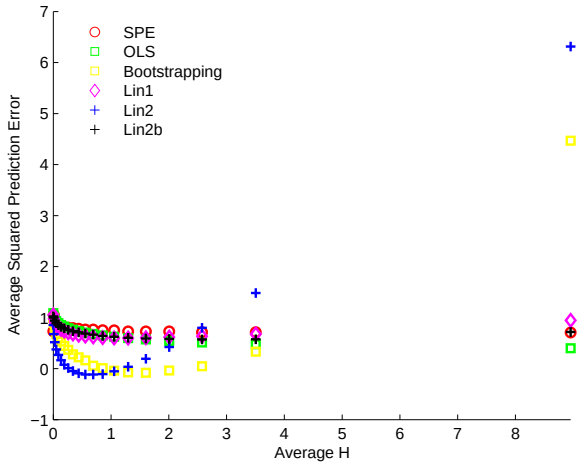


(b) average hb, Equation (5.12)

Figure 5.5: PLS Average Squared Prediction Error versus Average Distance Measure. $k = 2$, $a = 1$, $\mathrm{Var}\,(\dot{\mathbf{X}}_{c_1}) = 25$, $\mathrm{Var}\,(\dot{\mathbf{X}}_{c_2}) = 1$, $\beta_1 = 0$, $\beta_0 = \beta_2 = 1$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$. OLS: the ordinary least squares type prediction mean squared error, Equation (5.8). Bootstrapping: the bootstrapping-by-residual prediction mean squared error, Equation (5.10). Lin1: Denham's prediction mean squared error, Equation (5.13). Lin2: the new linearisation prediction mean squared error, Equation (5.16). Lin2b: the new linearisation prediction mean squared error embedded with bootstrapping, Equation (5.19).

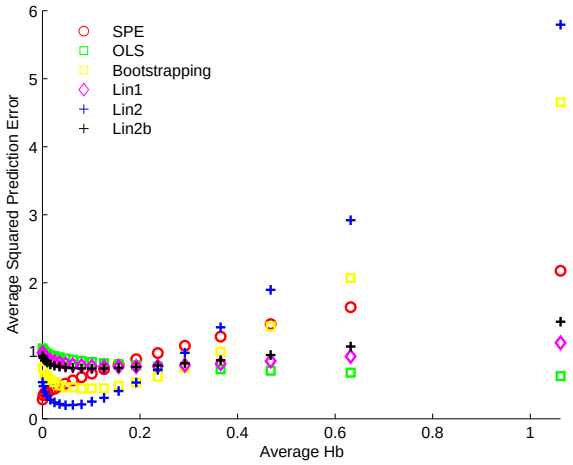


(c) average Hden, Equation (5.15)



(d) average H, Equation (5.18)

Figure 5.5: PLS Average Squared Prediction Error versus Average Distance Measure. $k = 2$, $a = 1$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_1}\right) = 25$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_2}\right) = 1$, $\beta_1 = 0$, $\beta_0 = \beta_2 = 1$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).



(e) average Hb, Equation (5.21)

Figure 5.5: PLS Average Squared Prediction Error versus Average Distance Measure. $k = 2$, $a = 1$, $\text{Var}\left(\dot{\mathbf{X}}_{c_1}\right) = 25$, $\text{Var}\left(\dot{\mathbf{X}}_{c_2}\right) = 1$, $\beta_1 = 0$, $\beta_0 = \beta_2 = 1$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).

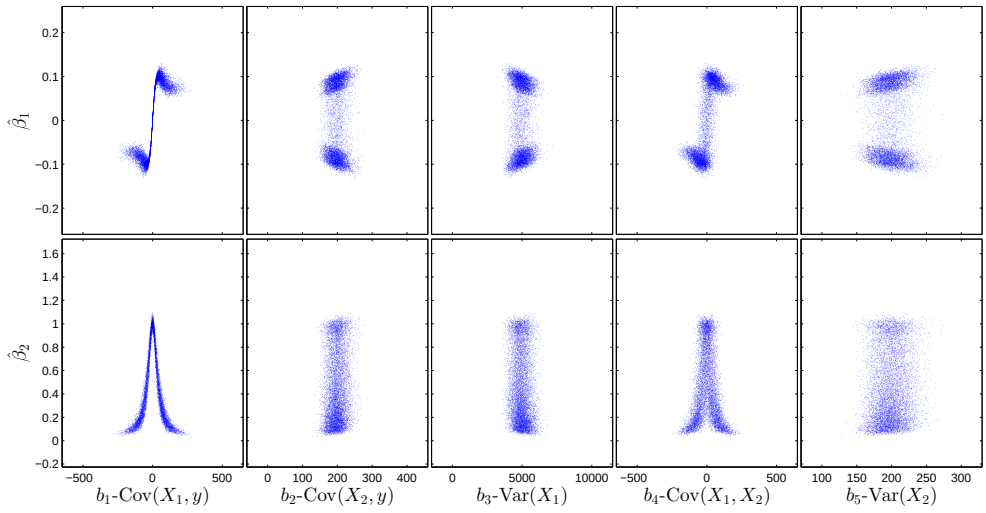
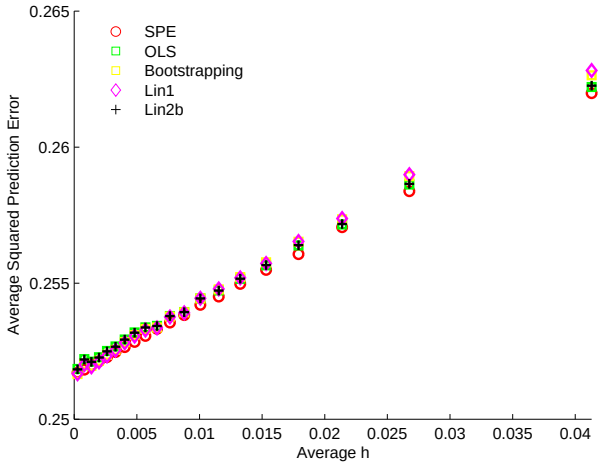
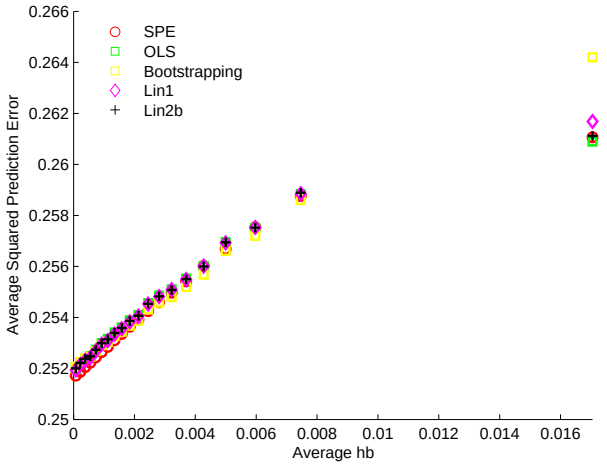


Figure 5.6: PLS $\hat{\boldsymbol{\beta}}$ against $\mathbf{b}$ when $k = 2$, $a = 1$, $\text{Var}\left(\dot{\mathbf{X}}_{c_1}\right) = 25$, $\text{Var}\left(\dot{\mathbf{X}}_{c_2}\right) = 1$, $\beta_1 = 0$, $\beta_0 = \beta_2 = 1$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$.

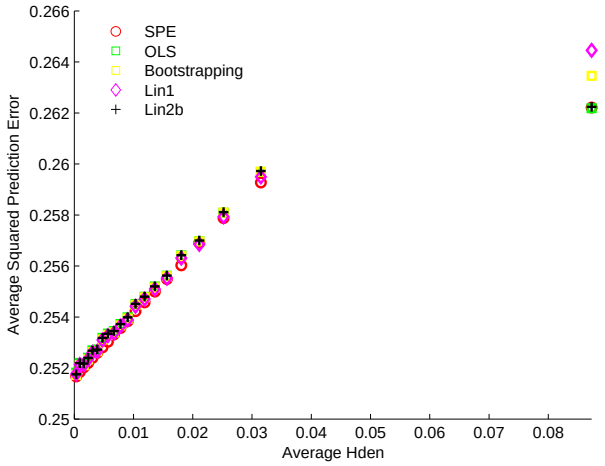


(a) average leverage h, Equation (5.9)

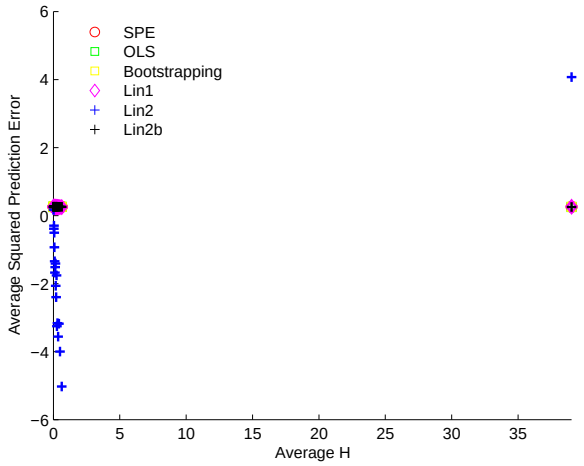


(b) average hb, Equation (5.12)

Figure 5.7: PLS Average Squared Prediction Error versus Average Distance Measure. $k = 3$, $a = 2$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_1}\right) = \mathrm{Var}\left(\dot{\mathbf{X}}_{c_2}\right) = 25$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_3}\right) = 1$, $\beta_1 = \beta_2 = 1$, $\beta_3 = 0$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$. OLS: the ordinary least squares type prediction mean squared error, Equation (5.8). Bootstrapping: the bootstrapping-by-residual prediction mean squared error, Equation (5.10). Lin1: Denham's prediction mean squared error, Equation (5.13). Lin2: the new linearisation prediction mean squared error, Equation (5.16). Lin2b: the new linearisation prediction mean squared error embedded with bootstrapping, Equation (5.19).

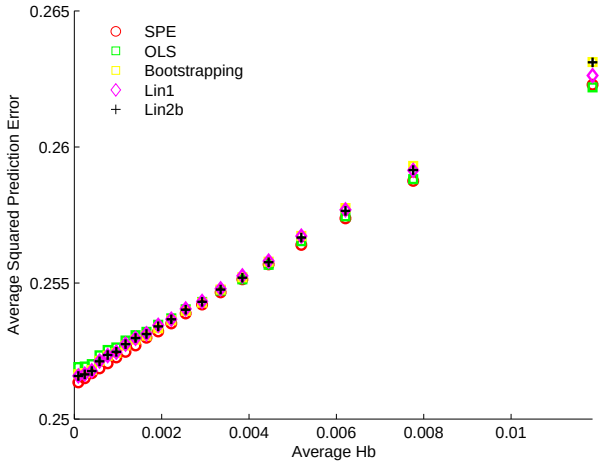


(c) average Hden, Equation (5.15)



(d) average H, Equation (5.18)

Figure 5.7: PLS Average Squared Prediction Error versus Average Distance Measure. $k = 3$, $a = 2$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_1}\right) = \mathrm{Var}\left(\dot{\mathbf{X}}_{c_2}\right) = 25$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_3}\right) = 1$, $\beta_1 = \beta_2 = 1$, $\beta_3 = 0$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).



(e) average Hb, Equation (5.21)

Figure 5.7: PLS Average Squared Prediction Error versus Average Distance Measure. $k = 3$, $a = 2$, $\text{Var}\left(\dot{\mathbf{X}}_{c_1}\right) = \text{Var}\left(\dot{\mathbf{X}}_{c_2}\right) = 25$, $\text{Var}\left(\dot{\mathbf{X}}_{c_3}\right) = 1$, $\beta_1 = \beta_2 = 1$, $\beta_3 = 0$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).

Figure 5.7(d) shows that Lin2 fails to give reasonable prediction mean squared error. The last blue + point is high above all other points, so it presses other 19 blue + points down to below zero as the whole line estimates squared prediction error on average. It is impossible to have negative squared prediction error, but the regression variance estimate Equation (5.17) may result in negative values when H is calculated to be quite large. This can be use as a sign warning the new linearisation method does not work. Therefore, Lin2 was taken away from Figure 5.7(a), (b), (c), and (e) because if the extreme large prediction mean squared error given by Lin2 expands the scales, the relationship presented by OLS, Bootstrapping, Lin1, and Lin2b would be unclear.

Figure 5.7(a), (b), (c), and (e) illustrates that OLS, Bootstrapping, Lin1 and Lin2b all work, except when h, hb, Hden and Hb become large. The last few points tend to over-estimate squared prediction error. Although Lin2 and Lin2b share the same idea, Lin2b calculates the variance of the estimated regression coefficients from the re-sampling, so it does not have the problem like Lin2, where the problem is caused by using the linear approximation to find the variance of the estimated

regression coefficients.

We are going to show how the new linearisation method (Lin2) fails using Figure 5.8 and Figure 5.9. Figure 5.8 gives the histograms of six selected elements

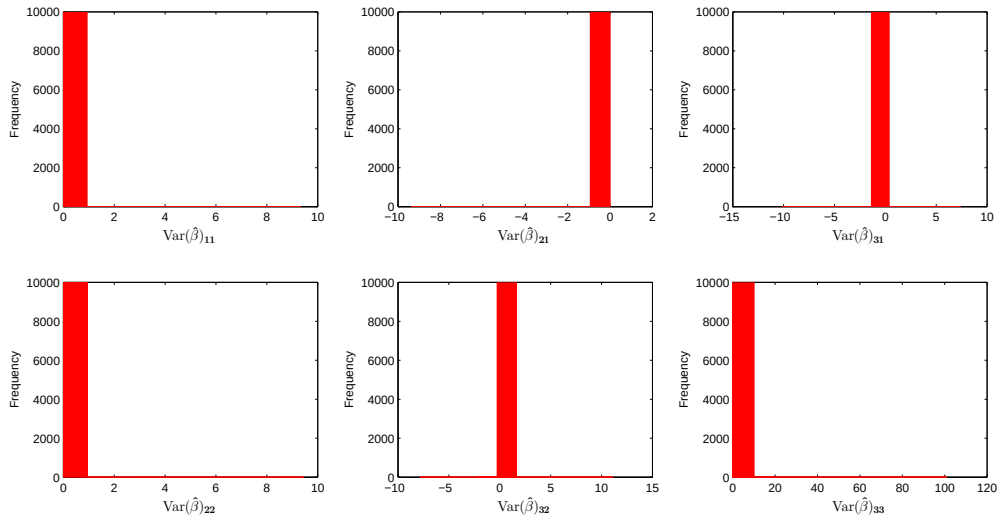


Figure 5.8: PLS Histograms for Six Selected Elements in Var $(\hat{\boldsymbol{\beta}})$ Calculated by the New Linearisation Method in the Case when $k = 3$, $a = 2$, Var $(\dot{\mathbf{X}}_{c_1})$ = Var $(\dot{\mathbf{X}}_{c_2}) = 25$, Var $(\dot{\mathbf{X}}_{c_3}) = 1$, $\beta_1 = \beta_2 = 1$, $\beta_3 = 0$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$. The subscript denotes the position of the element. For example, Var $(\hat{\boldsymbol{\beta}})_{21}$ represents the element at the second row and the first column of the variance matrix Var $(\hat{\boldsymbol{\beta}})$.

in the Var $(\hat{\boldsymbol{\beta}})$ matrix. The thin red line spanning on the sides of the main red bar says that the prediction mean squared error given by Lin2 is inflated heavily by some extreme values of Var $(\hat{\boldsymbol{\beta}}) = \left(\frac{\partial \hat{\boldsymbol{\beta}}}{\partial \mathbf{b}}\right)_{\mathbf{b}_0}$ Var $(\mathbf{b}_0)\left(\frac{\partial \hat{\boldsymbol{\beta}}}{\partial \mathbf{b}}\right)'_{\mathbf{b}_0}$. These extreme values appear when the new linearisation method fails to give a good approximation for some particular calibration sets.

Figure 5.9 gives an example of a calibration set that has extreme H values. It illustrates how well the new linearisation approximation method works for this calibration set. The blue dot points represent how the estimated regression coefficients $\hat{\beta}_1$, $\hat{\beta}_2$, and $\hat{\beta}_3$ actually vary with small changes of $\mathbf{b}$. The dash lines are linear approximations to the relationships between the estimated regression coefficients and $\Delta b_1, \Delta b_2, \ldots, \Delta b_9$. It can be seen that the relationships between

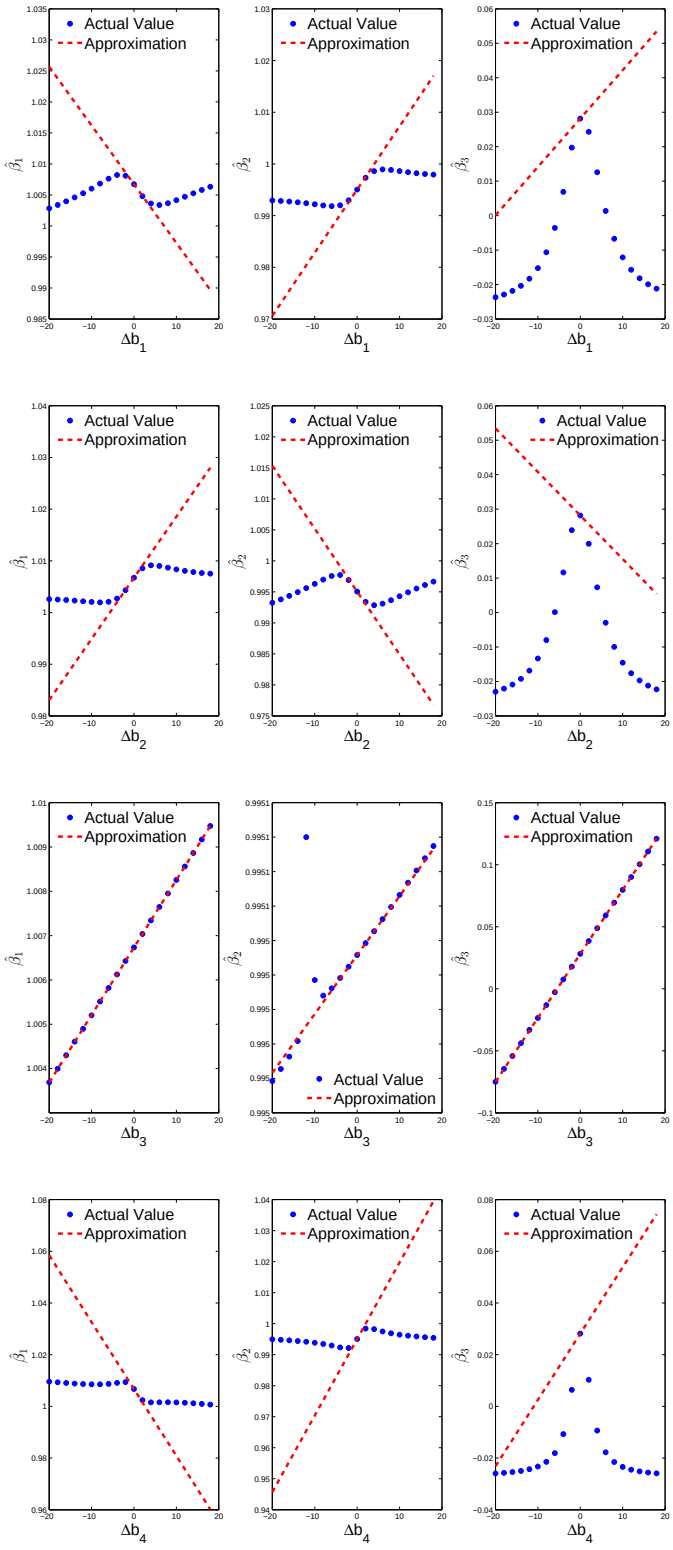


Figure 5.9: PLS Goodness of Fit for the New Linearisation Approximation in the Case when $k = 3$, $a = 2$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_1}\right) = \mathrm{Var}\left(\dot{\mathbf{X}}_{c_2}\right) = 25$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_3}\right) = 1$, $\beta_1 = \beta_2 = 1$, $\beta_3 = 0$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$.

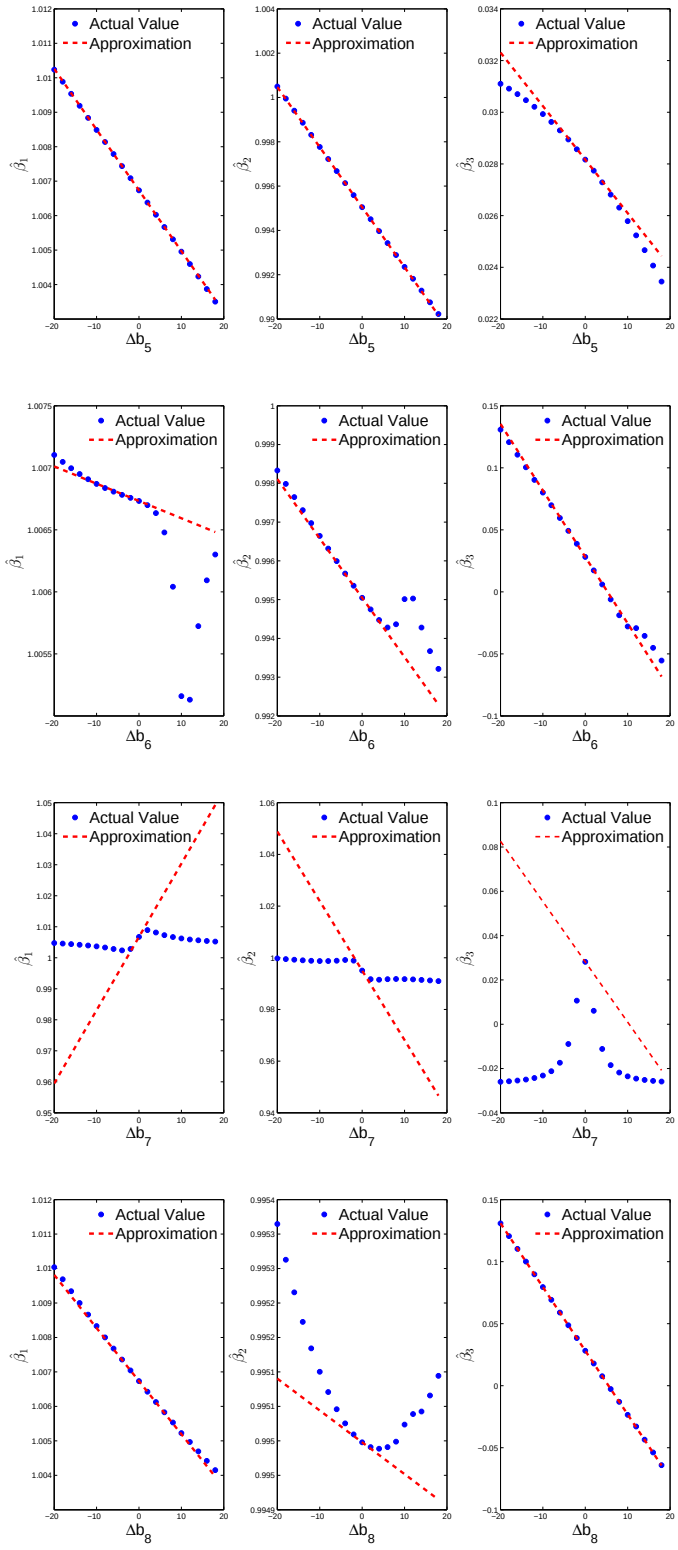


Figure 5.9: PLS Goodness of Fit for the New Linearisation Approximation in the Case when $k = 3$, $a = 2$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_1}\right) = \mathrm{Var}\left(\dot{\mathbf{X}}_{c_2}\right) = 25$, $\mathrm{Var}\left(\dot{\mathbf{X}}_{c_3}\right) = 1$, $\beta_1 = \beta_2 = 1$, $\beta_3 = 0$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).

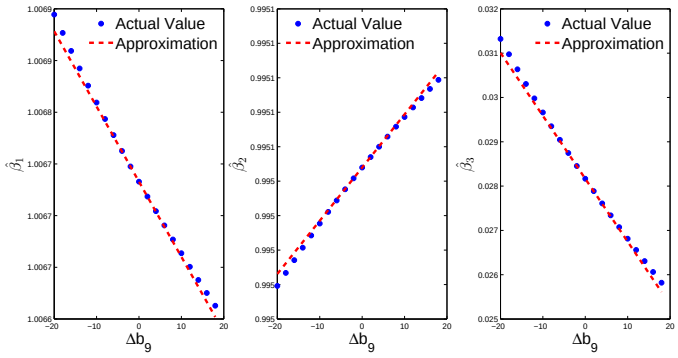


Figure 5.9: PLS Goodness of Fit for the New Linearisation Approximation in the Case when $k = 3$, $a = 2$, $\mathrm{Var}\,(\dot{\mathbf{X}}_{c_1}) = \mathrm{Var}\,(\dot{\mathbf{X}}_{c_2}) = 25$, $\mathrm{Var}\,(\dot{\mathbf{X}}_{c_3}) = 1$, $\beta_1 = \beta_2 = 1$, $\beta_3 = 0$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).

$\hat{\beta}_1$, $\hat{\beta}_2$, $\hat{\beta}_3$ and $\Delta\mathbf{b}$ are all poorly approximated. Specially, with small changes of b_1, b_2, b_4 and b_7, the new linear approximation is valid within a very narrow range. In the study of the relationships between $\hat{\beta}_1$, $\hat{\beta}_2$ and Δb_1, Δb_2, Δb_4, Δb_7, the 'z' shape of blue dot points shows the new linearisation method gives either too big or too small approximations on the two sides. In the plots of $\hat{\beta}_3$ against Δb_1, Δb_2, Δb_4,Δb_7, the blue dot points form a shape of 'Λ', saying that the approximate values are bigger than the true values except at the centre point. The 'U' shape of blue dot points in the plot of $\hat{\beta}_2$ against Δb_7 gives an example that the new linear approximation underestimates the rate of $\hat{\beta}_2$ changing with Δb_7.

Figure 5.10 is drawn to show how the linear approximation proposed by Denham (1997) works for the same calibration set. The linear approximation seems to give good estimates of how regression coefficients change with small disturbances in y_c. The red dash line overlaps with the blue dots line in the plots of $\hat{\beta}_1$ and $\hat{\beta}_1$ against Δy_c; the red line looks a bit more noisy in the plot of $\hat{\beta}_3$ against Δy_c, but it works all right. In comparison with Lin2, Lin1 gives sensible linear approximations for this particular calibration set.

Simulation 5.5. $k = 24$, $a = 7$, $\sigma_{c_1}^2 = \cdots = \sigma_{c_{24}}^2 = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24} = 1$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$, $N = 500$.

The simplest case when $k = 24$ and $a = 7$ is being discussed as it is the

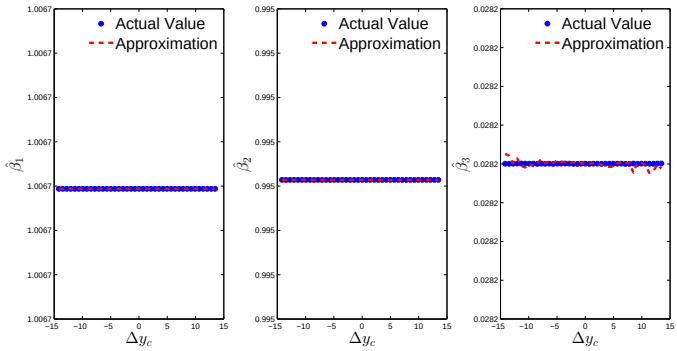
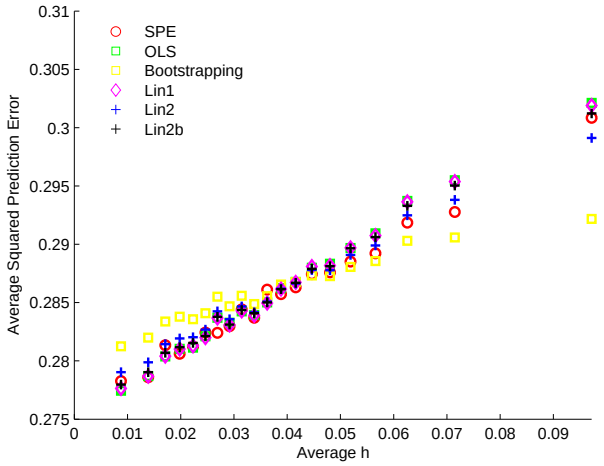


Figure 5.10: PLS Goodness of Fit for the Linear Approximation used by Denham (1997) in the Case when $k = 3$, $a = 2$, $\mathrm{Var}\,(\dot{\mathbf{X}}_{c_1}) = \mathrm{Var}\,(\dot{\mathbf{X}}_{c_2}) = 25$, $\mathrm{Var}\,(\dot{\mathbf{X}}_{c_3}) = 1$, $\beta_1 = \beta_2 = 1$, $\beta_3 = 0$, $\sigma_\epsilon^2 = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$.

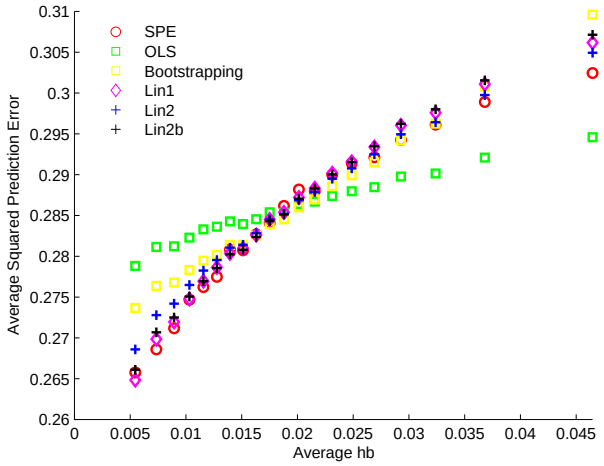
basic structure of the silage dataset in Section 5.6.1. The means and the standard errors of the estimated regression coefficients presented in Table 5.1 tell that the partial least squares regression is fitted properly. OLS, Lin1, and Lin2 seem to give reasonable prediction variances in Figure 5.11. Bootstrapping and Lin2b do not work so well as the other methods. When the number of explanatory variables becomes large, Bootstrapping and Lin2b tend not to behave well. Several similar simulations have been tried.

Table 5.1: PLS Means and Standard Errors of Estimated Regression Coefficients, $k = 24$, $a = 7$

	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$	$\hat{\beta}_5$	$\hat{\beta}_6$	$\hat{\beta}_7$	$\hat{\beta}_8$
mean	0.9995	0.9992	1.0034	0.9996	1.0025	1.0015	1.0014	0.9988
se	0.0381	0.0371	0.0357	0.0381	0.0373	0.0379	0.0395	0.0392
	$\hat{\beta}_9$	$\hat{\beta}_{10}$	$\hat{\beta}_{11}$	$\hat{\beta}_{12}$	$\hat{\beta}_{13}$	$\hat{\beta}_{14}$	$\hat{\beta}_{15}$	$\hat{\beta}_{16}$
mean	1.0013	1.0008	1.0009	1.0000	1.0001	0.9992	0.9985	0.9982
se	0.0380	0.0365	0.0413	0.0377	0.0389	0.0388	0.0381	0.0379
	$\hat{\beta}_{17}$	$\hat{\beta}_{18}$	$\hat{\beta}_{19}$	$\hat{\beta}_{20}$	$\hat{\beta}_{21}$	$\hat{\beta}_{22}$	$\hat{\beta}_{23}$	$\hat{\beta}_{24}$
mean	0.9994	1.0000	1.0026	1.0004	0.9981	1.0024	0.9984	0.9995
se	0.0362	0.0425	0.0371	0.0358	0.0382	0.0382	0.0368	0.0392

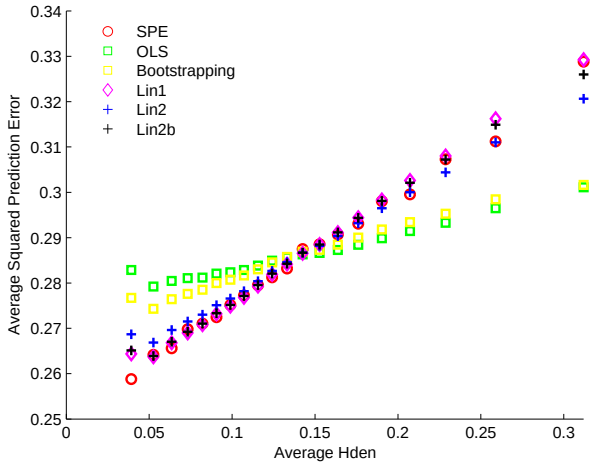


(a) average leverage h, Equation (5.9)

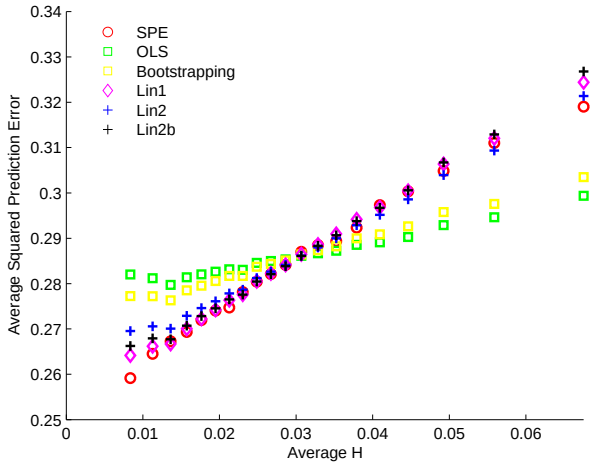


(b) average hb, Equation (5.12)

Figure 5.11: PLS Average Squared Prediction Error versus Average Distance Measure with $\hat{\sigma}^2_{\epsilon_t}$. $k = 24$, $a = 7$, $\mathrm{Var}(\dot{X}_{c_1}) = \cdots = \mathrm{Var}(\dot{X}_{c_{24}}) = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24}$, $\sigma^2_\epsilon = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$. OLS: the ordinary least squares type prediction mean squared error, Equation (5.8). Bootstrapping: the bootstrapping-by-residual prediction mean squared error, Equation (5.10). Lin1: Denham's prediction mean squared error, Equation (5.13). Lin2: the new linearisation prediction mean squared error, Equation (5.16). Lin2b: the new linearisation prediction mean squared error embedded with bootstrapping, Equation (5.19).

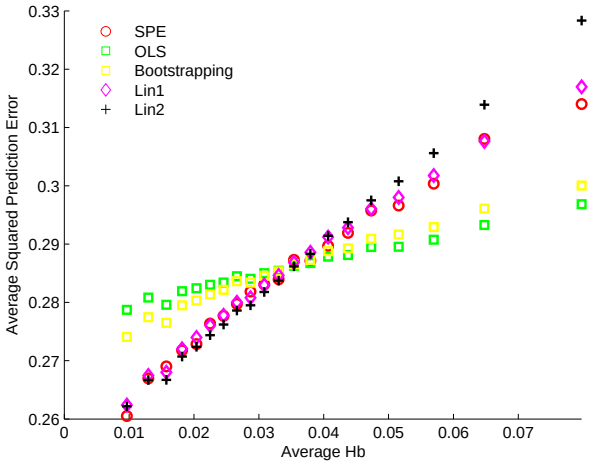


(c) average Hden, Equation (5.15)



(d) average H, Equation (5.18)

Figure 5.11: PLS Average Squared Prediction Error versus Average Distance Measure with $\hat{\sigma}^2_{\epsilon_t}$. $k = 24$, $a = 7$, $\mathrm{Var}\left(\dot{X}_{c_1}\right) = \cdots = \mathrm{Var}\left(\dot{X}_{c_{24}}\right) = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24}$, $\sigma^2_\epsilon = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).



(e) average Hb, Equation (5.21)

Figure 5.11: PLS Average Squared Prediction Error versus Average Distance Measure with $\hat{\sigma}^2_{\epsilon_t}$. $k = 24$, $a = 7$, $\mathrm{Var}\,(\dot{X}_{c_1}) = \cdots = \mathrm{Var}\,(\dot{X}_{c_{24}}) = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24}$, $\sigma^2_\epsilon = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).

The slopes of the yellow square point line (Bootstrapping) and the green square point line (OLS) are quite different in Figure 5.11(b). It notifies that the prediction mean squared error goes up faster in the direction of hb, when hb and the part of h projecting onto the direction of hb both increase one unit. Similarly, it can be explained that the square green and yellow point lines (OLS and Bootstrapping) are flatter than the others in Figure 5.11(c) - (e).

5.5.2 Partial Least Squares Regression Simulation Using the Estimated Regression Error Variance from the Calibration Set

In Section 5.5.1 we employ the estimated regression error variance from the tuning set, which is suggested by the results shown in Section 3.3.1 principal components regression simulation that the regression error variance estimate from the tuning set compensates the omission of unused components. To demonstrate why the estimated regression error variance from the tuning set is better, in this section we shall plot the result of Simulation 5.5 using $\hat{\sigma}^2_{\epsilon_c}$, Equation (4.17), calculated from

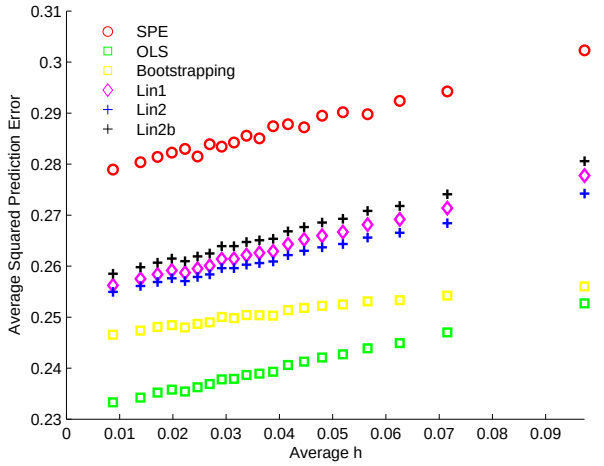
the calibration set, in Figure 5.11(c). $\hat{\sigma}^2_{\epsilon_c}$ uses the number of factors $a+1$ as a simple estimate of degrees of freedom, hence it underestimates the regression error variance. These prediction mean squared error formulae are presented in Section 5.4.

In Figure 5.12(a) there is a gap between the red point line (SPE) and the green square point line (OLS). This is consistent with principal components regression theory that the unselected components cause the bias, where the simulation example is presented in Figure 3.2 of Section 3.3.1. In partial least squares regression, the new factors are built by a linear combination of explanatory variables, where the unused parts of explanatory variables become the omitted components, that cause the so-called bias. Section 3.3.2 has verified that the expected squared bias is the difference between SPE and OLS, which also applied in univariate partial least squares regression. In the ordinary least squares type prediction mean squared error formula, $\hat{\sigma}^2_{\epsilon_c}$ controls the slope and the intercept of the ordinary least squares type prediction mean squared error. It under-estimates the true regression variance, so OLS seems parallel with and below SPE.

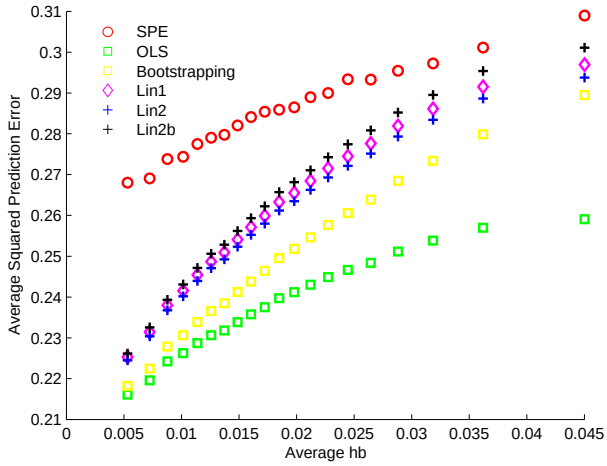
In Figure 5.12(c) the Denham's prediction mean squared error (Lin1) looks also parallel with SPE. It is because in the Denhams's prediction mean squared error formula, the intercept and the slope equal to $\hat{\sigma}^2_{\epsilon_c}$, although the calculation of Hden is estimated with respect to the small change of $\mathbf{y}_c$, which automatically involves these unused components.

Figure 5.12(b), (d), and (e) shows the intercepts of Bootstrapping, Lin2 and Lin2b are all smaller than the actual values, but the slopes of Bootstrapping, Lin2 and Lin2b looks steeper than that of SPE. In the bootstrapping by residual prediction mean squared error formula, hb, calculated from $\mathrm{Var}\,(\hat{\boldsymbol{\beta}}^B)$, includes the estimation of the variations about the bias during the re-sampling. In the new linearisation method H and Hb are obtained with respect to the sums of squares of explanatory variables and response variable, so the unused components are being used.

When we use the estimated regression error variance from the tuning drawn

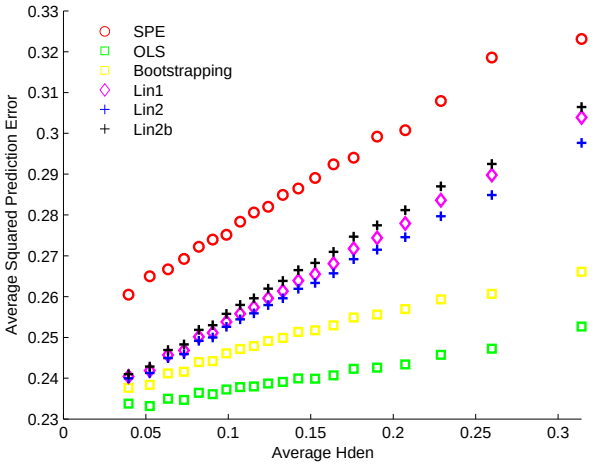


(a) average leverage h, Equation (5.9)

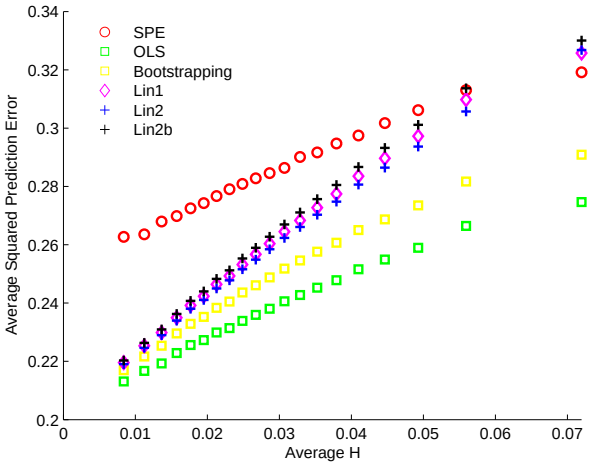


(b) average hb, Equation (5.12)

Figure 5.12: PLS Average Squared Prediction Error versus Average Distance Measure with $\hat{\sigma}^2_{\epsilon_c}$. $k = 24$, $a = 7$, $\mathrm{Var}\,(\dot{X}_{c_1}) = \cdots = \mathrm{Var}\,(\dot{X}_{c_{24}}) = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24} = 1$, $\sigma^2_\epsilon = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$. OLS: the ordinary least squares type prediction mean squared error, Equation (5.8). Bootstrapping: the bootstrapping-by-residual prediction mean squared error, Equation (5.10). Lin1: Denham's prediction mean squared error, Equation (5.13). Lin2: the new linearisation prediction mean squared error, Equation (5.16). Lin2b: the new linearisation prediction mean squared error embedded with bootstrapping, Equation (5.19).

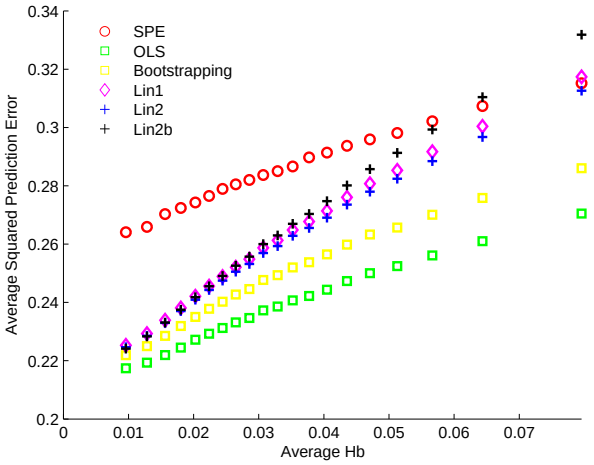


(c) average Hden, Equation (5.15)



(d) average H, Equation (5.18)

Figure 5.12: PLS Average Squared Prediction Error versus Average Distance Measure with $\hat{\sigma}^2_{\epsilon_c}$. $k = 24$, $a = 7$, $\mathrm{Var}\left(\dot{X}_{c_1}\right) = \cdots = \mathrm{Var}\left(\dot{X}_{c_{24}}\right) = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24} = 1$, $\sigma^2_\epsilon = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).



(e) average Hb, Equation (5.21)

Figure 5.12: PLS Average Squared Prediction Error versus Average Distance Measure with $\hat{\sigma}^2_{\epsilon_c}$. $k = 24$, $a = 7$, $\mathrm{Var}(\dot{X}_{c_1}) = \cdots = \mathrm{Var}(\dot{X}_{c_{24}}) = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24} = 1$, $\sigma^2_\epsilon = 0.25$, $\boldsymbol{\epsilon}_p = \mathbf{0}$ (cont.).

in Figure 5.11, the slope of Bootstrapping, Lin2 and Lin2b become flatter. To explain what happen, we take Lin2 as an example, Equation (5.17) gives

$$\hat{\sigma}^2_{\epsilon_{l2}} = \frac{\frac{1}{n_t}\sum_{j=1}^{n_t}\{(\dot{y}_{t_j} - \hat{\dot{y}}_{t_j})^2 - \mathrm{H}_{t_j}\}}{1 + \frac{1}{n_t}},$$

which can be rearranged as,

$$= \frac{1}{n_t + 1}\sum_{j=1}^{n_t}(\dot{y}_{t_j} - \hat{\dot{y}}_{t_j})^2 - \frac{1}{n_t + 1}\sum_{j=1}^{n_t}\mathrm{H}_{t_j}.$$

$\frac{1}{n_t+1}\sum_{j=1}^{n_t}(\dot{y}_{t_j} - \hat{\dot{y}}_{t_j})^2$ is equivalent to an adjusted $\hat{\sigma}^2_{\epsilon_c}$. The average term, $\frac{1}{n_t+1}\sum_{j=1}^{n_t}\mathrm{H}_{t_j}$, cancels out parts of H in the new linearisation prediction mean squared error formula, Equation (5.2) in Section 5.4,

$$\mathrm{E}\{(\dot{y}_p - \hat{\dot{y}}_p)^2\} = \hat{\sigma}^2_{\epsilon_{l2}}(1 + \frac{1}{n}) + \mathrm{H}.$$

Hence, the slope of Lin2 decreases when we use the estimated regression error variance from the tuning set. The same reason applies to Bootstrapping and Lin2b.

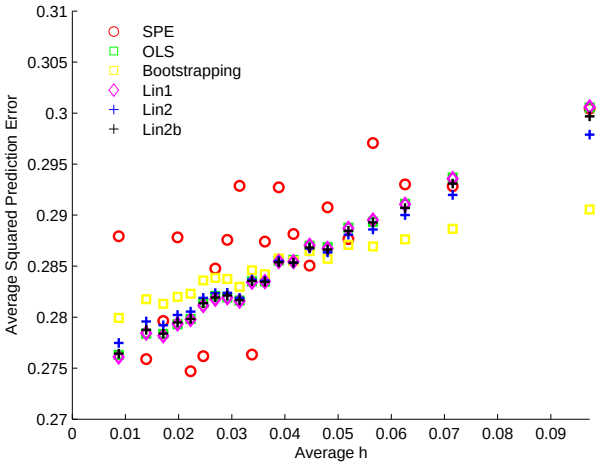
The analysis suggests the estimated regression error variance from the tuning set should be a powerful tool to ensure these prediction mean squared error formulae to be right on average.

5.5.3 Partial Least Squares Regression Simulation Including the Error Term in Prediction Samples

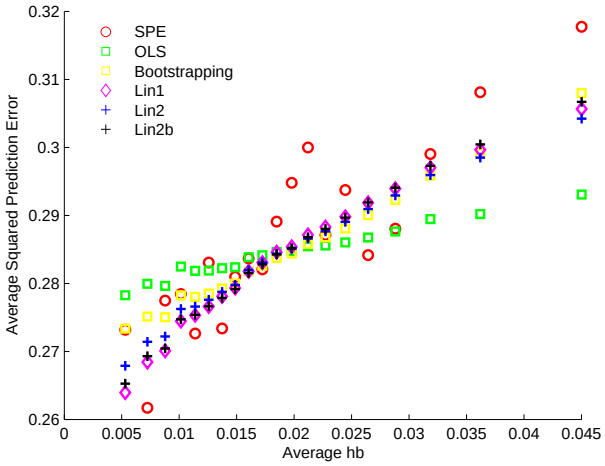
Simulation 5.6. $k = 24$, $a = 7$, $\sigma^2_{c_1} = \cdots = \sigma^2_{c_{24}} = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24} = 1$, $\sigma^2_\epsilon = 0.25$, $N = 500$.

Different from Simulation 5.5, Simulation 5.6 includes the error term in the simulation. Without the noise free assumption, this simulation resembles the analysis with a lots of real data sets. The result is stored in Figure 5.13. Compared to Figure 5.11, the red points in Figure 5.13, average squared prediction error (SPE), are more noisy, which is specially obvious in Figure 5.13(a). With the increasing noise, the ordinary least squares type prediction mean squared error (OLS) with the assistance of the estimated regression error variance from the tuning set still describes the average behavior of SPE. Figure 5.13(b) - (e) demonstrate the bootstrapping by residual prediction mean squared error (Bootstrapping), the Denhams's prediction mean squared error (Lin1) and the new linearisation method (Lin2) and (Lin2b) work too.

The ordinary least squares type prediction mean squared error only considers the variation in $\mathbf{X}_c'\mathbf{X}_c$. It does not contain the variation about $\mathbf{y}_c$, so it relies completely on the estimated regression error variance to compensate the bias. Comparatively, the bootstrapping by residual prediction mean squared error, the Denham's prediction mean squared error, the new linearisation prediction mean squared error and its bootstrapping realisation use the whole information provided by explanatory variables and response variable together in their mathematical mechanisms, to find the variance of estimated regression coefficients, which can be regarded as taking the bias into account. In other words, they absorb the estimation of the noise variation into the measurement of hb, Hden, H and Hb. Although they need the estimated regression error variance to make the average

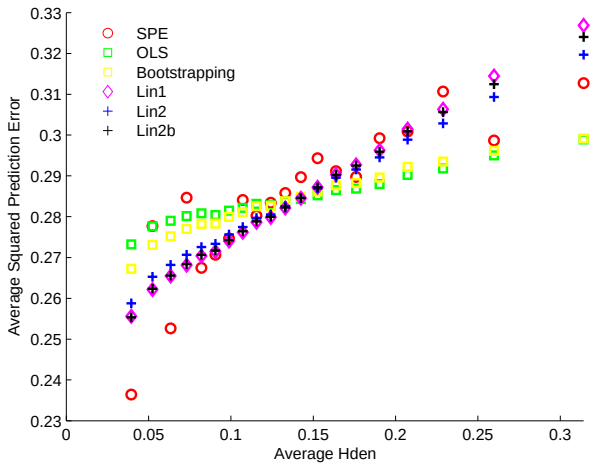


(a) average leverage h, Equation (5.9)

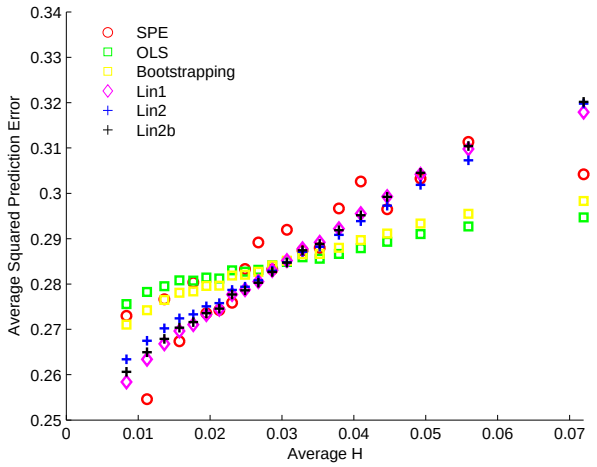


(b) average hb, Equation (5.12)

Figure 5.13: PLS Average Squared Prediction Error versus Average Distance Measure with $\hat{\sigma}^2_{\epsilon_t}$. $k = 24$, $a = 7$, $\mathrm{Var}(\dot{X}_{c_1}) = \cdots = \mathrm{Var}(\dot{X}_{c_{24}}) = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24} = 1$, $\sigma^2_\epsilon = 0.25$, $\boldsymbol{\xi}_p \neq \mathbf{0}$. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$. OLS: the ordinary least squares type prediction mean squared error, Equation (5.8). Bootstrapping: the bootstrapping-by-residual prediction mean squared error, Equation (5.10). Lin1: Denham's prediction mean squared error, Equation (5.13). Lin2: the new linearisation prediction mean squared error, Equation (5.16). Lin2b: the new linearisation prediction mean squared error embedded with bootstrapping, Equation (5.19).

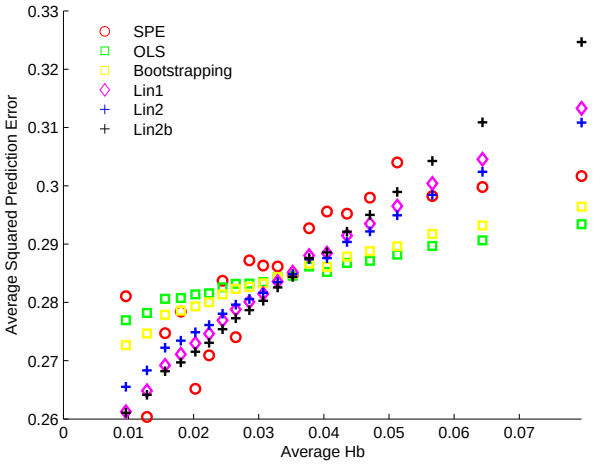


(c) average Hden, Equation (5.15)



(d) average H, Equation (5.18)

Figure 5.13: PLS Average Squared Prediction Error versus Average Distance Measure with $\hat{\sigma}^2_{\epsilon_t}$. $k = 24$, $a = 7$, $\mathrm{Var}\left(\dot{X}_{c_1}\right) = \cdots = \mathrm{Var}\left(\dot{X}_{c_{24}}\right) = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24} = 1$, $\sigma^2_\epsilon = 0.25$, $\boldsymbol{\xi}_p \neq \mathbf{0}$ (cont.).



(e) average Hb, Equation (5.21)

Figure 5.13: PLS Average Squared Prediction Error versus Average Distance Measure with $\hat{\sigma}^2_{\epsilon_t}$. $k = 24$, $a = 7$, $\mathrm{Var}(\dot{X}_{c_1}) = \cdots = \mathrm{Var}(\dot{X}_{c_{24}}) = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24} = 1$, $\sigma^2_\epsilon = 0.25$, $\boldsymbol{\xi}_p \neq \mathbf{0}$ (cont.).

right, but they are not so sensitive to the noise like OLS. This is why in Figure 5.13(b) -(e) Bootstrapping, Lin1, Lin2 and Lin2b seem to fit SPE more tightly than OLS does.

The analysis confirms again that the use of the estimated regression variance from the tuning set is worthwhile.

5.6 An Example of Real Data Analysis

5.6.1 Silage Data Analysis

To investigate the performance of the new local linearisation method, a silage dataset used in Fearn and Davies (2003) has been employed to investigate squared prediction errors. The silage dataset comprised 774 samples that had been produced in 1994 and 1995, and were characterised in 1995 by conventional silage analytical techniques and near infrared measurements at the Institute of Grassland and Forage Research at FAL, Braunschweig, Germany. These samples were originally from dairy farms in North Germany. Each sample has a 100-point near

infrared transmission spectrum from 850 to 1048 nm in 2 nm steps, measured using an Infratec Meat Analyser 1265. Ammonia nitrogen (NH_3NTM) was measured as a characteristic indicator of the protein breakdown during silage fermentation, and it was determined using an ammonia-sensitive electrode and related to the dry matter of the sample. After missing reference measurements and an outlier are removed from the sample, there are left with 692 samples for ammonia nitrogen. The Savitsky Golay second derivative of the spectrum is used in the analysis, which comprises 94 points, and then an equal-step 24 points are truncated from the second derivative of the spectrum. Because the new linearisation method is computationally expensive, a smaller number of explanatory variables would save time. The explanatory variables are 24 second derivative of the spectrum, and the response variable is the ammonia nitrogen in terms of dry matter in percentage. Using leave-twenty-out cross-validation we choose the number of factors of partial least squares regression as $a = 7$ (See Figure 5.14).

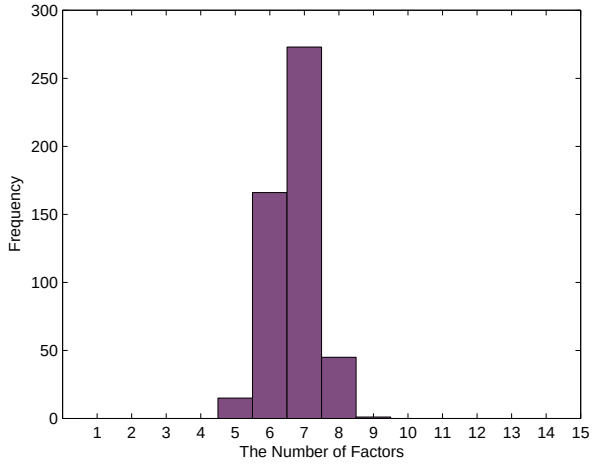


Figure 5.14: PLS Histogram: the Number of Factors, Silage Data

From these 692 samples we randomly select 200 observations as the calibration set and 100 observations as the tuning set, then use the remaining 392 observations as the prediction set. The random data splitting procedure runs 500 times. Figure 5.15 gives the analysis results. None of OLS, Bootstrapping, Lin1, Lin2 and Lin2b seems to work appropriately for the silage data. Despite of the noise in the plots,

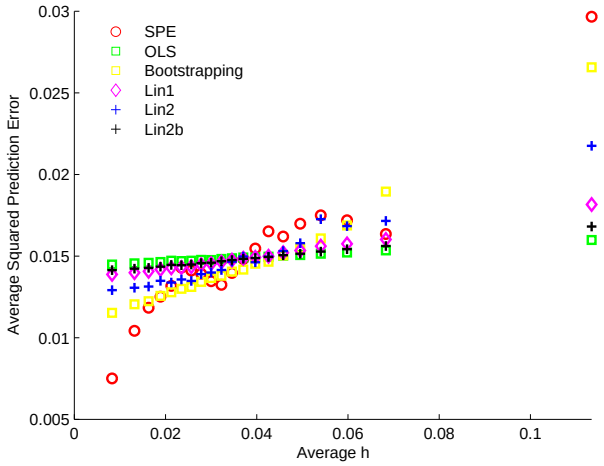
there is a difference between the general trend of prediction mean squared error estimates and the actual relationship of squared prediction error and distance metrics. We will discuss why they do not work.

In Figure 5.15(a), the red circle point line (SPE) is noisy, and it is non-linear. The first few SPE points seems to drop down, and the last SPE point is high away from the others. It tells that there are extreme squared prediction errors when the leverage is either very small or very large. SPE is steeper than the green square point line (OLS). In Figure 5.15(c) the magenta diamond points (Lin1) has a similar relationship with the red circle points (SPE), so does the black plus point line (Lin2b) in Figure 5.15(e).

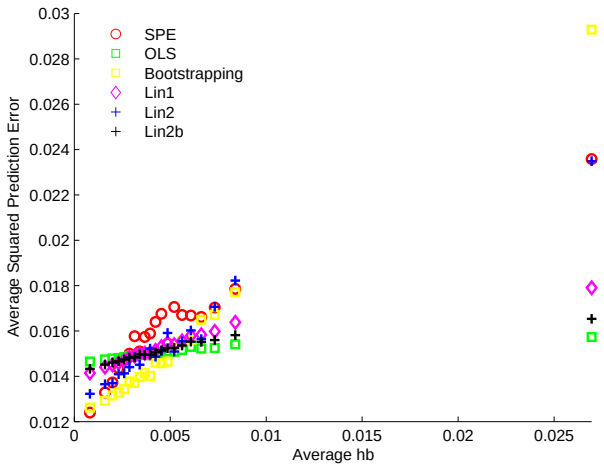
In Figure 5.15(b), the last red circle point (SPE) is far away its general trend, which is again the evidence there exist extreme large hb values. The yellow square point line (Bootstrapping) is below SPE, except its last point that stays higher than the last point of SPE. This is similar to that displayed in Figure 5.15(d), where the last blue plus point (Lin2) looks like a star hanging in the sky. All other 19 points of Lin2 are under the red circle points (SPE).

We have understood in Section 5.4 that prediction variances presented by OLS, Bootstrapping, Lin1, Lin2 and Lin2b estimate squared prediction error on average, due to the use of the estimated regression variance from the tuning set. This is the reason why all lines seem to cross in the middle in Figure 5.15(a).

However, the adjusted regression variance estimate may inflate or lessen the slope of the prediction mean squared error line when there exists extreme distance measure values. In Figure 5.15(d), the last Lin2 point is large, so the average adjustment of the estimated regression variance pushes down the other 19 Lin2 points. This reveals there may exist very large prediction variances produced by large H values, which is similar to Simulation 5.4, where Lin2 does not give a good linear approximation for some particular calibration sets. This can be confirmed by the histograms of six selected elements of Var $(\hat{\boldsymbol{\beta}})$ displayed in Figure 5.16. The fine red lines on the two sides of the main bar show evidence of the extreme values. These extreme values also make the intersection of all lines shifting backward in

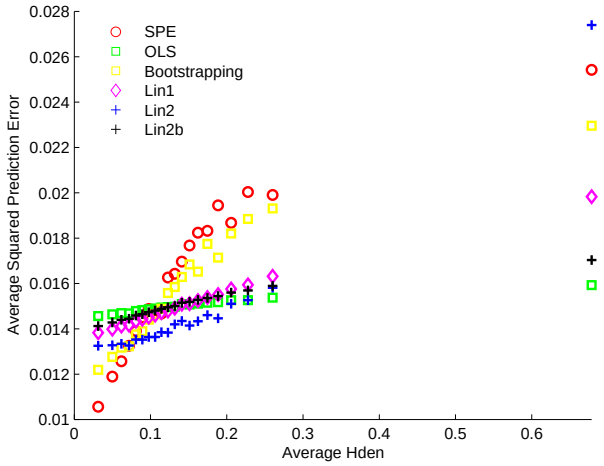


(a) average leverage h, Equation (5.9)

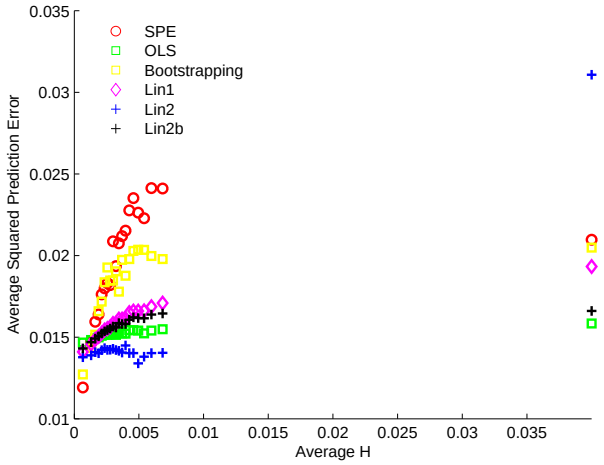


(b) average hb, Equation (5.12)

Figure 5.15: PLS Average Squared Prediction Error versus Average Distance Measure, Silage Data. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$. OLS: the ordinary least squares type prediction mean squared error, Equation (5.8). Bootstrapping: the bootstrapping-by-residual prediction mean squared error, Equation (5.10). Lin1: Denham's prediction mean squared error, Equation (5.13). Lin2: the new linearisation prediction mean squared error, Equation (5.16). Lin2b: the new linearisation prediction mean squared error embedded with bootstrapping, Equation (5.19).

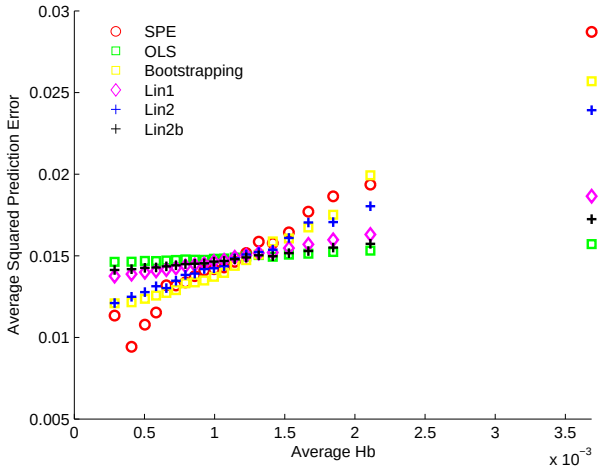


(c) average Hden, Equation (5.15)



(d) average H, Equation (5.18)

Figure 5.15: PLS Average Squared Prediction Error versus Average Distance Measure, Silage Data (cont.).



(e) average Hb, Equation (5.21)

Figure 5.15: PLS Average Squared Prediction Error versus Average Distance Measure, Silage Data (cont.).

Figure 5.15(c) and (e), or moving forward in Figure 5.15(b) and (d) because the regression variance estimate from the tuning set adjusts prediction mean squared error to be right on average.

Table 5.2 presents the means and the standard errors of the estimated regression coefficients. The big standard error is another evidence that the partial least squares regression do not work properly for the silage data, so extreme values may appear in the calculations of prediction mean squared error.

The existence of these extreme values in the calculation of prediction variances is not the only reason that all the methods seem to fail. The other reason, which is the main reason, is because there is only one dataset, which is similar to the ordinary least squares regression simulation study (Simulation 2.1), Equation (2.6) notes that for a fixed calibration set the relationship between squared prediction error and leverage may not be linear. And, Simulation 2.4 shows the random data splitting systematically repeats the noise in the fixed dataset. Hence, it is not surprising that SPE points in Figure 5.15 do not present a linear relationship, no matter with the leverage or other measure metrics. As the same as the ordinary least squares prediction variance, these prediction mean squared error formulae

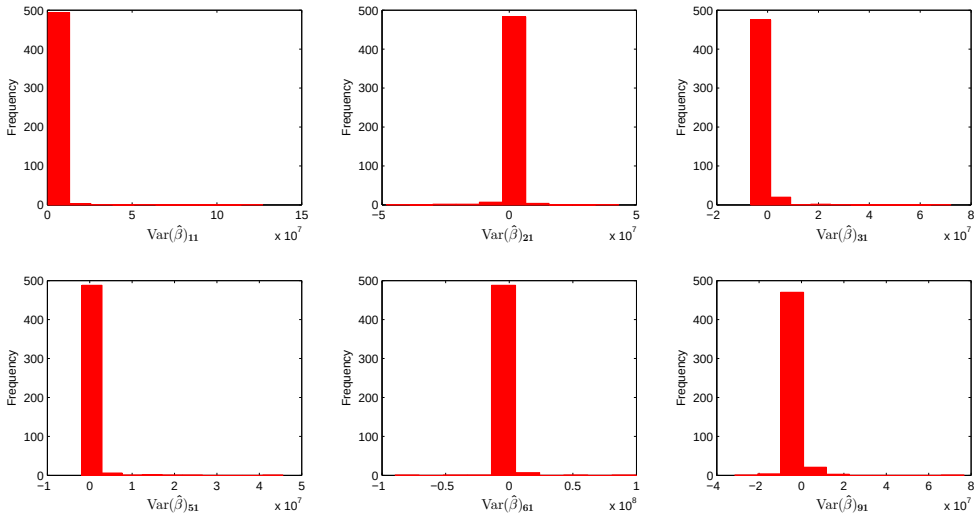


Figure 5.16: PLS Histogram for Six Selected Elements of Var $(\hat{\boldsymbol{\beta}})$ Calculated by the New Linearisation Method, Silage Data. The subscript denotes the position of the element. For example, Var $(\hat{\boldsymbol{\beta}})_{51}$ represents the element at the fifth row and the first column of the variance matrix Var $(\hat{\boldsymbol{\beta}})$.

are all expectations over repeatedly sampled calibration sets. Therefore, similarly to the result given by Simulation 2.3, the performance of these prediction mean squared error formulae cannot be assessed in an obvious way referring to the fixed dataset. Although the random data splitting allows us to estimate the distribution of the estimated regression coefficients, this distribution is biased, so the squared prediction error calculated for a single set of data always relies on these biased estimated regression coefficients from this dataset.

If the analysis is right, the logic applies to any single dataset. Hence, we will analyse a simulated dataset using random data splitting in next section, where all conditions are set to be exactly the same as the silage dataset.

Table 5.2: PLS Means and Standard Errors of Estimated Regression Coefficients, Silage Data

	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$	$\hat{\beta}_5$	$\hat{\beta}_6$	$\hat{\beta}_7$	$\hat{\beta}_8$
mean	143.0	580.1	1739.4	-2792.2	1097.3	3843.9	577.0	-2092.8
se	1113.3	845.1	560.4	1048.7	1242.4	1211.2	1042.4	753.7
	$\hat{\beta}_9$	$\hat{\beta}_{10}$	$\hat{\beta}_{11}$	$\hat{\beta}_{12}$	$\hat{\beta}_{13}$	$\hat{\beta}_{14}$	$\hat{\beta}_{15}$	$\hat{\beta}_{16}$
mean	-4118.8	-2127.0	1528.8	3424.5	354.2	-341.3	-1734.1	543.9
se	1175.6	1266.8	576.4	1061.7	439.6	471.6	949.7	496.2
	$\hat{\beta}_{17}$	$\hat{\beta}_{18}$	$\hat{\beta}_{19}$	$\hat{\beta}_{20}$	$\hat{\beta}_{21}$	$\hat{\beta}_{22}$	$\hat{\beta}_{23}$	$\hat{\beta}_{24}$
mean	4284.5	2071.8	103.1	-1206.7	-2034.3	-940.2	-1197.3	-1812.5
se	770.5	889.7	707.0	485.2	691.0	943.8	714.3	821.7

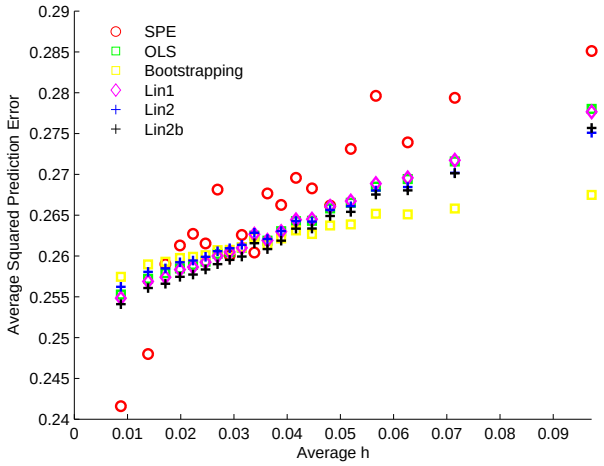
5.6.2 Random Data Splitting Simulation in Imitation of the Silage Data

Simulation 5.7. The Investigation of Random Data Splitting when $k = 24$, $a = 7$, $\sigma_{c_1} = \cdots = \sigma_{c_{24}} = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24} = 1$, $\sigma_\epsilon^2 = 0.25$.

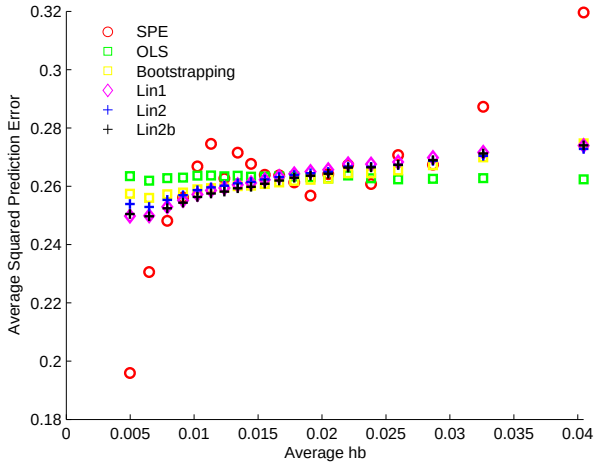
To examine the analysis of the silage data, we run 500 random splits of a particular simulated dataset of 692 observations. The analysis is conducted exactly as the same as the silage data. The results are drawn in Figure 5.17.

Similar to Figure 5.15, the red points (SPE) does not have a linear trend in Figure 5.17(a). In Figure 5.17(b)-(e) SPE points forms a loose 'Z' pattern. Despite the noisy, OLS, Bootstrapping, Lin1, Lin2, and Lin2b cross the middle of SPE lines, but none of them provide a good estimate of SPE. The results suggests that there is a problem to assess the performance of these prediction mean squared error formulae, which is as the same as the analysis of the silage data.

Table 5.3 shows that the means and the standard errors of estimated regression coefficients. They seem reasonable, which suggests that the partial least squares regression is fitted properly, so there is no problem in the calculation of Lin2. This can be confirmed again from the histograms of six elements of $\mathrm{Var}\,(\hat{\boldsymbol{\beta}})$ in Figure

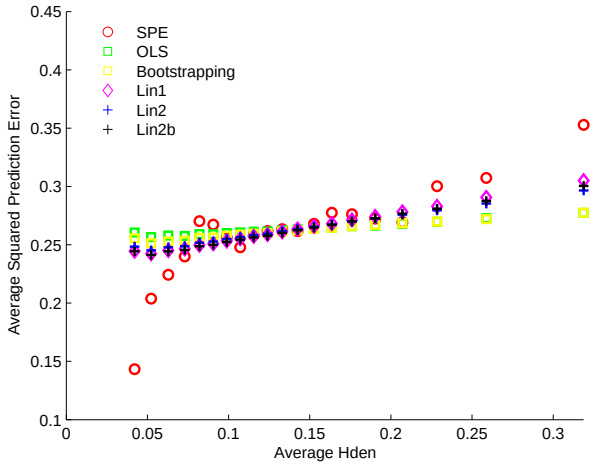


(a) average leverage h, Equation (5.9)

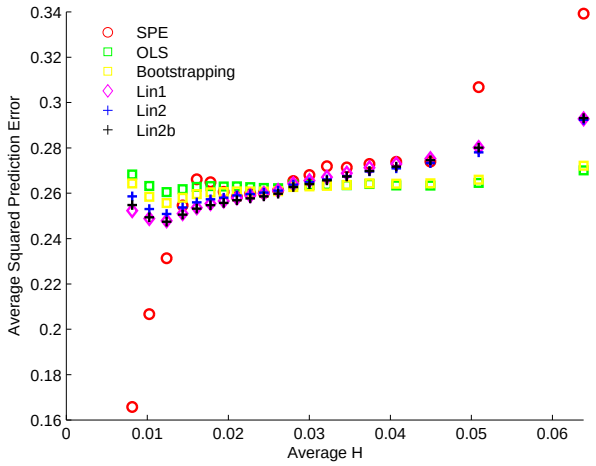


(b) average hb, Equation (5.12)

Figure 5.17: PLS Average Squared Prediction Error versus Average Distance Measure, random data splitting. $k = 24$, $a = 7$, $\mathrm{Var}\,(\dot{X}_{c_1}) = \cdots = \mathrm{Var}\,(\dot{X}_{c_{24}}) = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24} = 1$, $\sigma_\epsilon^2 = 0.25$. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$. OLS: the ordinary least squares type prediction mean squared error, Equation (5.8). Bootstrapping: the bootstrapping-by-residual prediction mean squared error, Equation (5.10). Lin1: Denham's prediction mean squared error, Equation (5.13). Lin2: the new linearisation prediction mean squared error, Equation (5.16). Lin2b: the new linearisation prediction mean squared error embedded with bootstrapping, Equation (5.19).

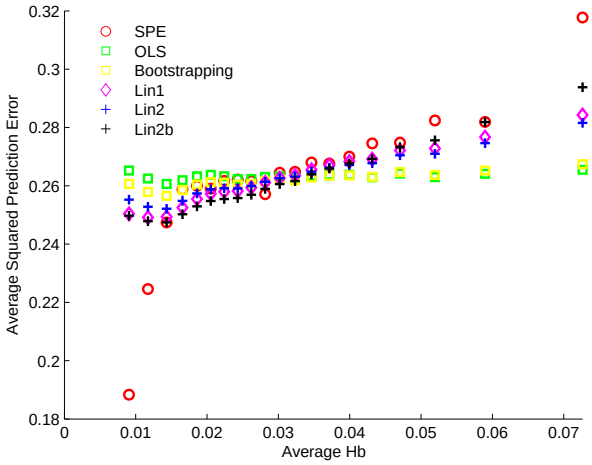


(c) average Hden, Equation (5.15)



(d) average H, Equation (5.18)

Figure 5.17: PLS Average Squared Prediction Error versus Average Distance Measure, random data splitting. $k = 24$, $a = 7$, $\mathrm{Var}\left(\dot{X}_{c_1}\right) = \cdots = \mathrm{Var}\left(\dot{X}_{c_{24}}\right) = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24} = 1$, $\sigma_\epsilon^2 = 0.25$ (cont.).



(e) average Hb, Equation (5.21)

Figure 5.17: PLS Average Squared Prediction Error versus Average Distance Measure, random data splitting. $k = 24$, $a = 7$, $\text{Var}\,(\dot{X}_{c_1}) = \cdots = \text{Var}\,(\dot{X}_{c_{24}}) = 1$, $\beta_0 = \beta_1 = \cdots = \beta_{24} = 1$, $\sigma_\epsilon^2 = 0.25$ (cont.).

5.18. All histograms look like normal distributions. The x-axis scale spans 10^{-3} or 10^{-4}, which is much smaller than 10^7 or 10^8, the x-axis scale shown in Figure 5.16 for the silage data, even the two graphs concern the same elements of $\text{Var}\,(\hat{\boldsymbol{\beta}})$.

To make the assessment problem clear, we will run simulations using the simple case $k = a = 1$ in Section 5.6.3 to show that the random data splitting gives a biased distribution of the estimated regression coefficients.

5.6.3 Simple Random Data Splitting Simulations

Simulation 5.8. Random Data Splitting for One Set of Simulated Data when $k = a = 1$

For one dataset, the estimated regression coefficients are systematically different from the true regression coefficients, which is unknown. To see how the discrepancy between the expected value and the result of a real data, the random data splitting simulation in the case when $k = 1$ and $a = 1$ is investigated. A dataset $(\dot{\mathbf{x}}_{c_0}, \dot{\mathbf{y}}_{c_0})$ with 692 observations is generated, where $\dot{\mathbf{x}}_{c_0}$ is independent identically normally distributed with mean zero and variance 4. $\dot{\mathbf{y}}_{c_0} = \beta_0 + \beta\dot{\mathbf{x}}_{c_0} + \boldsymbol{\epsilon}$,

Table 5.3: PLS Means and Standard Errors of Estimated Regression Coefficients, $k = 24$, $a = 7$, random data splitting.

	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$	$\hat{\beta}_5$	$\hat{\beta}_6$	$\hat{\beta}_7$	$\hat{\beta}_8$
mean	1.0230	1.0110	0.9524	1.0011	0.9798	1.0281	0.9856	0.9916
se	0.0331	0.0336	0.0298	0.0338	0.0321	0.0304	0.0309	0.0290
	$\hat{\beta}_9$	$\hat{\beta}_{10}$	$\hat{\beta}_{11}$	$\hat{\beta}_{12}$	$\hat{\beta}_{13}$	$\hat{\beta}_{14}$	$\hat{\beta}_{15}$	$\hat{\beta}_{16}$
mean	0.9997	1.0210	1.0291	1.0251	1.0546	0.9947	1.0139	0.9840
se	0.0340	0.0347	0.0321	0.0341	0.0301	0.0333	0.0333	0.0329
	$\hat{\beta}_{17}$	$\hat{\beta}_{18}$	$\hat{\beta}_{19}$	$\hat{\beta}_{20}$	$\hat{\beta}_{21}$	$\hat{\beta}_{22}$	$\hat{\beta}_{23}$	$\hat{\beta}_{24}$
mean	0.9779	1.0311	1.0248	1.0363	1.0063	1.0096	0.9582	1.0161
se	0.0316	0.0288	0.0319	0.0320	0.0337	0.0296	0.0300	0.0326

where regression coefficients $\beta_0 = \beta = 1$, and the noise term $\boldsymbol{\epsilon}$ follows the standard normal distribution. The averages over 500 random replicates of the simulated data are studied. In each random split, a calibration set of 200 observations, a tuning set of 100 observations, and a prediction set of 392 observations are drawn randomly from $(\dot{\mathbf{X}}_{c_0}, \dot{\mathbf{y}}_{c_0})$, which is the same as the silage data.

The result is presented in Figure 5.19. The blue points (SPE) are the average squared prediction error against average leverage for the partial least squares regression. The solid red line (SPE OLSfit) is the ordinary least squares fit of all squared prediction error against leverage for the partial least squares regression, which overlaps with the pink dot line (SPEols OLSfit) that gives the ordinary least squares fit of all squared prediction error against leverage for the ordinary least squares regression.

We fit ordinary least squares regression to the simulated data too. The pink dash-dot line (SPEols OLSfit) denotes the ordinary least squares fit of squared prediction error against leverage for the ordinary least squares regression.

When $k = a = 1$, partial least squares regression is equivalent to ordinary least squares regression. This has been confirmed by the fact that SPE OLSfit and SPEols OLSfit overlap.

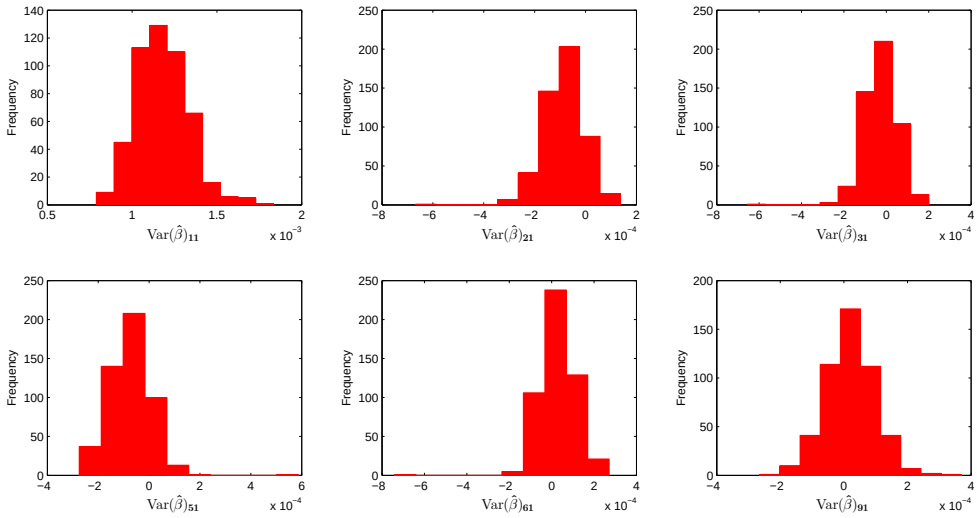


Figure 5.18: PLS Histogram for Six Selected Elements of Var $(\hat{\boldsymbol{\beta}})$ Calculated by the New Linearisation Method, random data splitting, $k = 24$, $a = 7$. The subscript denotes the position of the element. For example, Var $(\hat{\boldsymbol{\beta}})_{51}$ represents the element at the fifth row and the first column of the variance matrix Var $(\hat{\boldsymbol{\beta}})$.

The bootstrapping by residuals method and the new linearisation method are time-consuming. The bootstrapping version of the new linearisation method gives similar result. Hence, only the ordinary least squares type expression (OLS), the Denham's linearisation method (Lin1) and the new linearisation method bootstrapping version (Lin2b) are included in the simulation.

In the case of $k = a = 1$, it has been shown in Simulation 5.1 that the ordinary least squares type prediction mean squared error, the classical linearsiation prediction mean squared error, and the new linearisation method prediction mean squared error are approximately the same. It explains Lin1 and Lin2b perfectly lie on OLS.

Similarly to the green points in Figure 2.5 of Simulation 2.4 in Section 2.2, blue points are noisy, and the slope of SPE OLSfit is quite different from that of OLS. The noise demonstrates that the random data splitting systematically repeats the variations in the dataset. The slope of SPE OLSfit can be bigger, smaller or equal to that of OLS, because there is only one set of simulated data.

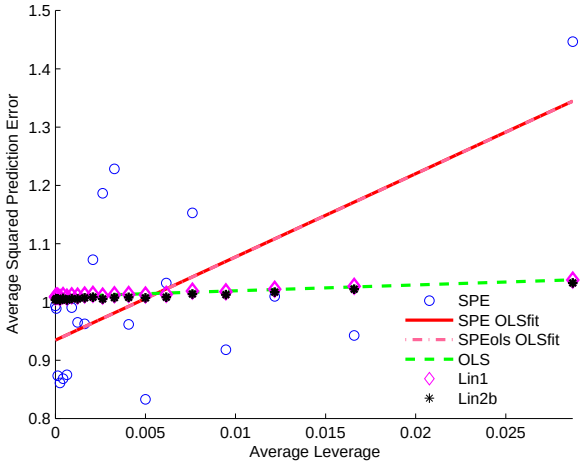


Figure 5.19: PLS Average Squared Prediction Error against Average Leverage for One Set of Simulated Data, random data splitting, $k = a = 1$. SPE: average squared prediction error $(\dot{y}_p - \hat{\dot{y}}_p)^2$ in the partial least squares regression. SPE OLSfit: the ordinary least squares fit of all squared prediction error in the partial least squares regression. SPEols OLSfit: the ordinary least squares fit of all squared prediction errors in the ordinary least squares regression. OLS: the ordinary least squares type partial least squares prediction mean squared error, Equation (5.8). Bootstrapping: the bootstrapping-by-residual prediction mean squared error, Equation (5.10). Lin1: Denham's prediction mean squared error, Equation (5.13). Lin2b: the new linearisation prediction mean squared error embedded with bootstrapping, Equation (5.19).

The ordinary least squares type prediction mean squared error takes expectations over the distribution of estimated regression coefficients.

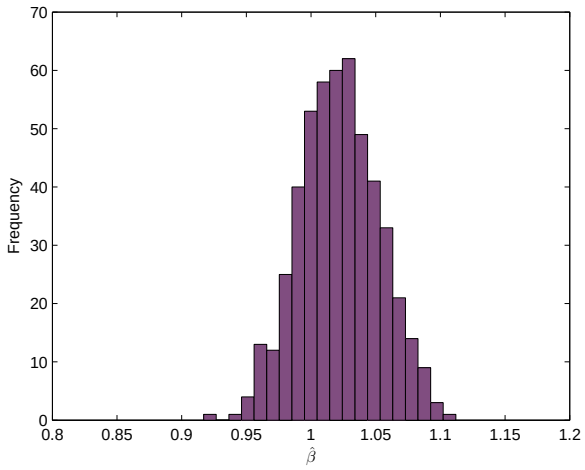


Figure 5.20: PLS Histogram: $\hat{\beta}$ for One Set of Simulated Data, random data splitting, $k = a = 1$.

The mean of $\hat{\beta} = 1.0219$, and the histogram of $\hat{\beta}$ is shown in Figure 5.20. From the histogram, we can easily see a systematic bias, where $\hat{\beta}$ has a normal distribution with mean 1.0219, but the true value of β is 1. A series of simulations have been tried. Hence, the results of Simulation 5.8 are in line with the regression theory verified in Simulation 2.4 in Section 2.2.5. Applying random data splitting in partial least squares regression and in ordinary least squares regression gives a biased distribution of the estimated regression coefficients. In order to see the expected values of the true regression coefficients, we use a simulation with $50,000$ sets of dataset to carry out random data splitting in next simulation.

Simulation 5.9. Random Data Splitting for $50,000$ Sets of Simulated Data when $k = a = 1$.

To compare with Simulation 5.8, we run the simulation of $50,000$ replicates, each of which generates a set of data, and then applies random data splitting analysis. We keep using ordinary least squares regression to fit the data as it is equivalent to partial least squares regression in this case. Since there are a

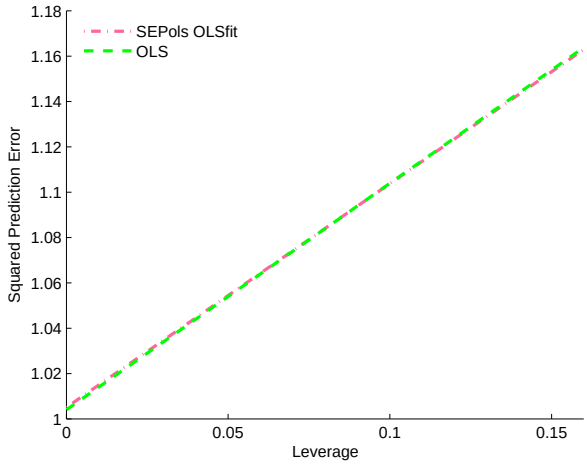


Figure 5.21: PLS Squared Prediction Error against Leverage for 50,000 Sets of Simulated Data, random data splitting, $k = a = 1$. SPEols OLSfit: the ordinary least square fit of all squared prediction errors in the ordinary least squares regression. OLS: the ordinary least squares type partial least squares prediction mean squared error.

large number of replicates, only the ordinary least squares type prediction mean squared error is applied in order to speed up the simulation. It is also because we have seen from Figure 5.19 that the prediction variances given by the ordinary least squares type expression, the Denham's linearisation method and the new linearisation method are approximately the same.

The simulation result is drawn in Figure 5.21. SPEols OLSfit denotes the ordinary least squares fit of all squared prediction error against leverage for the ordinary least square regression. OLS presents for the ordinary least squares type prediction mean squared error for partial least squares regression. SPEols OLSfit and OLS overlap, which agrees with Figure 2.6 of Simulation 2.5 in Section 2.2.5.

The mean of $\hat{\beta} = 0.9998$. It is very close to the true value of β. The histogram of $\hat{\beta}$, Figure 5.22, has a normal distribution with mean 0.9998. Hence, the results of Simulation 5.9 show evidence that the ordinary least squares type predication variance is the result of taking expectation over the distribution of estimated regression coefficients, so does the other prediction mean squared error

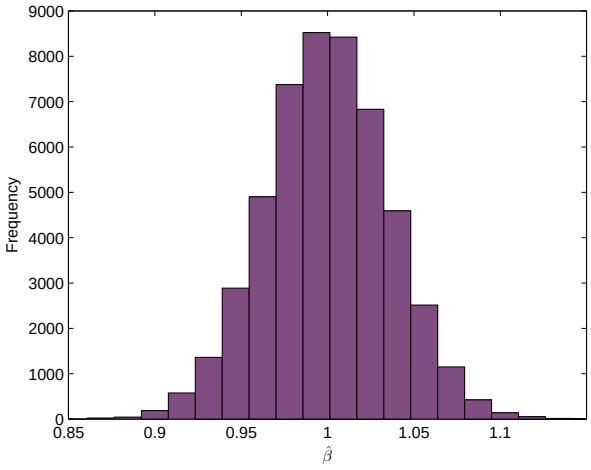


Figure 5.22: PLS Histogram: $\hat{\beta}$ for $50,000$ Sets of Simulated Data, random data splitting, $k = a = 1$.

formulae.

5.7 Summary

The noise free simulation studies, running over a large number of replicates that contains different calibration sets and prediction sets, suggests the prediction mean squared error formulae given by the ordinary least squares type expression, bootstrapping by residuals, the Denham's linearisation method, the new linearisation method and its bootstrapping realisation, all work with an estimated regression error variance calculated from the tuning set when the partial least squares regression is fitted properly, but sometimes the new linearisation method is unstable.

The simulation studies of using the estimated regression error variance from the tuning set in the prediction mean squared error formulae with the error term in the prediction set suggest that it should be wise to use the regression error variance estimates calculated from the tuning set.

Using the estimated regression error variance from the tuning set, the ordinary least squares type prediction mean squared error can be used as a parsimonious estimate of partial least squares prediction uncertainty. If we pursue a more deli-

cate result, the linearisation based methods would be recommended. However, the mathematics of the new linearisation method is complicated; it is computationally expensive; and sometimes it fails due to some special structure in the variance of explanatory variables. The Denham's linearisation method is more stable.

The analysis of the real dataset and its relevant simulations suggests it is impossible to evaluate the performance of the prediction mean squared error formulae on any real dataset. The problem of using the ordinary least squares regression prediction mean squared error formula on a single dataset described in Section 2.4, still cannot be solved by tackled random data splitting. These prediction mean squared error formulae take expectations over the distribution of the estimated regression coefficients, so the prediction mean squared error calculated from the random data splitting depends on the estimated regression coefficients from the full dataset. The random error of estimated regression coefficients causes a bias when the estimated regression coefficients are used in the prediction mean squared error formulae, hence it is disappointing these methods cannot be evaluated for a real dataset.

5.8 Appendix

5.8.1 $\partial \mathbf{S}_i \mathbf{w}_i / \partial \boldsymbol{vecut}(\mathbf{S}_i)$

$$\frac{\partial \mathbf{S}_i \mathbf{w}_i}{\partial vecut(\mathbf{S}_i)} = \begin{pmatrix} w_{i1} & 0 & 0 & \cdots & 0 \\ w_{i2} & w_{i1} & 0 & \cdots & 0 \\ w_{i3} & 0 & w_{i1} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{ik} & 0 & 0 & \cdots & w_{i1} \\ 0 & w_{i2} & 0 & \cdots & 0 \\ 0 & w_{i3} & w_{i2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & w_{ik} & 0 & \cdots & w_{i2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & w_{ip} \end{pmatrix}'.$$

5.8.2 $\partial \mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i / \partial \boldsymbol{vecut}(\mathbf{S}_i)$

As shown in Equation (5.7) of Section 5.2.2.1

$$\begin{aligned} \frac{\partial \mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i}{\partial \mathbf{S}_i} &= 2\mathbf{w}_i \mathbf{w}_i' - diag(\mathbf{w}_i \mathbf{w}_i')' \mathbf{I} \\ &= \begin{pmatrix} w_{i1}^2 & 2w_{i1}w_{i2} & \cdots & 2w_{i1}w_{ik} \\ 2w_{i2}w_{i1} & w_{i2}^2 & \cdots & 2w_{i2}w_{ik} \\ \vdots & \vdots & \ddots & \vdots \\ 2w_{ik}w_{i1} & 2w_{ik}w_{i2} & \cdots & w_{ik}^2 \end{pmatrix}, \end{aligned}$$

$$\frac{\partial \mathbf{w}_i' \mathbf{S}_i \mathbf{w}_i}{\partial vecut(\mathbf{S}_i)} = \begin{pmatrix} w_{i1}^2 & 2w_{i1}w_{i2} & \cdots & 2w_{i1}w_{ik} & w_{i2}^2 & \cdots & w_{ik}^2 \end{pmatrix}.$$

5.8.3 $\partial \boldsymbol{vecut}(\mathbf{u}_i\mathbf{u}_i')/\partial \mathbf{u}_i$

For the i-th iteration, let

$$\mathbf{u}_i = \begin{pmatrix} u_1 & u_2 & u_3 & \cdots & u_k \end{pmatrix}',$$

$$\frac{\partial vecut(\mathbf{u}_i\mathbf{u}_i')}{\partial \mathbf{u}_i} = \begin{pmatrix} 2u_1 & 0 & 0 & \cdots & 0 \\ u_2 & u_1 & 0 & \cdots & 0 \\ u_3 & 0 & u_1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ u_k & 0 & 0 & \cdots & u_1 \\ 0 & 2u_2 & 0 & \cdots & 0 \\ 0 & u_3 & u_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & u_k & 0 & \cdots & u_2 \\ 0 & 0 & 2u_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 2u_k \end{pmatrix}.$$

Chapter 6

Conclusions

In this chapter, we will briefly review all the work presented in this thesis, and then we will draw some conclusions about the quantification of prediction uncertainty for principal components regression and partial least squares regression. Possible future directions for research will also be discussed.

6.1 Ordinary Least Squares Regression Prediction Uncertainty Study Summary

Chapter 2 provides a foundation for studying principal components regression and partial least squares regression. There we review ordinary least squares regression theory and carry out various numerical experiments using this the simple scenario to give a guidance on the direction and the design of simulations for the study of principal components regression and partial least squares regression.

- We use simulation studies to verify the ordinary least squares prediction variance formula, which gives a linear relationship between squared prediction error and leverage. The simulation studies deepen our understanding of the ordinary least squares prediction variance formula, which takes expectation over the distribution of the estimated regression coefficients. For a fixed data set, it is noted that the relationship between squared prediction error and

leverage may not be linear and may be very different to that given by the formula.

- The simulation studies point out the performance of the ordinary least squares prediction variance formula cannot be judged on on any fixed real data set.

- We investigate the use of a tuning set and cross-validation, looking for a practical way to estimate prediction variance for a real data set.

  - The tuning set and the cross-validation are used to calculate root mean squared error of prediction (RMSEP) and root mean squared error of cross-validation (RMSECV), which are simple empirical estimates of prediction error.

  - The tuning set and the cross-validation also can be used to calculate estimated regression error variances, which can be plugged into the ordinary least squares prediction formula. This will be useful for principal components regression and partial least squares regression.

  - For a fixed real data set, it seems reasonable in theory to use the tuning set, or cross-validation, to directly model an approximate linear relationship between prediction variance and leverage, but this requires that the tuning set has a large sample size. It is difficult or not economic to collect so many samples, so this empirical method is infeasible in practice.

- Random data splitting is not helpful to round this assessment problem for a real data set, as it systematically repeats the variations in the data set.

The last step of principal components regression and partial least squares regression carries out an ordinary least squares regression regressing the response variable on the constructed factors. The problems found in the study of ordinary least squares prediction variance also exist in the quantification of the prediction

uncertainty for principal components regression and partial least squares regression.

6.2 Principal Components Regression Prediction Uncertainty Study Summary

Chapter 3 studies principal components regression.

- Besides the empirical estimates of prediction uncertainty, root mean squared error of prediction (RMSEP) and root mean squared error of cross-validation (RMSECV), the ordinary least squares type prediction mean squared error is an alternative for the estimation of prediction uncertainty.

- We use mathematics and simulation studies to show that the bias is a key player in the prediction uncertainty estimation for principal components regression.

- We propose an adjustment for the ordinary least squares type prediction mean squared error formula, which employs an estimated regression error variance from the tuning set to compensate the omission of the expected squared bias from the original formula.

- Inspired by the linear relationship between prediction mean squared error and leverage shown in the ordinary least squares type prediction mean squared error formula, we try to build simple linear models upon $\frac{1}{n}$ for the intercept and the slope of the ordinary type prediction mean squared error formula, since both prediction mean squared error and leverage are related to $\frac{1}{n}$. This empirical approach fails because further study suggests the slope is not linear with $\frac{1}{n}$.

- It is difficult to formulate a mathematical relationship between leverage and expected squared bias. The mathematics is complicated, so it is not usable to quantify prediction uncertainty.

6.3 Partial Least Squares Regression Prediction Uncertainty Study Summary

We start to study partial least squares regression from Chapter 4.

- We present two equivalent algorithms used in the thesis: orthogonal scores algorithms and orthogonal loadings algorithms.
- We study existing approaches in the literature to quantify partial least squares prediction uncertainty.
  - Like ordinary least squares regression and principal components regression, the simple empirical estimates of prediction error, root mean squared error of prediction (RMSEP) and root mean squared error of cross-validation (RMSECV) are useful, but they attach the same prediction error to all prediction samples.
  - The ordinary least squares type prediction mean squared error formula which omits the expected squared bias is widely used. It only considers the variations in the response variables.
  - A series of approaches are based on the ordinary least squares type prediction mean squared error, such as the linearisation methods, the re-sampling methods (bootstrapping by objects, bootstrapping by residuals, and jackknife). The re-sampling methods consider the variations in both of explanatory and response variables.
  - The classical linearisation method is proposed by Denham (1997) involves the linearisation of the partial least squares estimators. It also only considers the variations in the response variables. An alternative is put forward by Romera (2010). It gives us a new idea to explore the partial least squares prediction mean squared error in Chapter 5. It takes both of explanatory and response variables into account.
  - A so-called U-deviation method used by the chemometrics software Unscrambler and its related methods have also been studied. Although the

U-deviation method is not correct, and the other methods need to be further checked, they are valuable for bringing up the idea of studying the x-residual that measures the distance between the predictor and its projection on the new factor space. We tries to reveal the relationship between squared prediction error and the x-residual, but there is nothing obvious observed, so it is not reported here.

In Chapter 5 we present a new linearisation method on the basis of the idea proposed by Romera (2010). It carries out the linearisation of estimated regression coefficients with respect to the small changes of the covariance between explanatory and response variables as well as the variance of explanatory variables. Following this idea, we build a bootstrapping algorithm to realise the new linearisation method, in the pursuit of less expensive computation. We use simulation studies and real data analysis to compare the new linearisation method and its bootstrapping application with the ordinary least squares type prediction mean squared error, bootstrapping by residuals, and the Denham's linearisation method. The estimated regression error variance from the tuning set is employed as it compensates the omission of the bias to some extent.

6.4 Conclusions

- For principal components regression, the ordinary least squares type prediction mean squared error with the estimated regression error variance from the tuning set is a good alternative to the empirical estimates of prediction uncertainty, root mean squared error of prediction (RMSEP) and root mean squared error of cross-validation (RMSECV). The estimated regression error variance adjusts the ordinary least squares type prediction mean squared error formula for the omission of the bias.

- For partial least squares regression, in addition to the methods above, the bootstrapping by residuals, the Denham's linearisation method, and the new linearsiation method, with the help of the estimated regression error variance

from the tuning set, can all give reasonable estimates for prediction uncertainty. The new linearisation method is complicated and computationally expensive. The Denham's method is more stable than the proposed new linearisation method.

The ordinary least squares type prediction mean squared error with the estimated regression error variance from the tuning set works well. Even in the worst case where the new linearisation method fails, it still gives a reasonable result. Hence, it is advisable to use the ordinary least squares type prediction mean squared error with the estimated regression error variance from the tuning set because it is a sensible and cheap way to quantify partial least squares regression prediction uncertainty. To estimate regression error variance from the tuning set compensates the omission of the bias in the ordinary least squares type prediction mean squared error formula.

Just as in ordinary least squares regression, the performance of these prediction mean squared error formulae for principal components regression and partial least squares regression cannot be evaluated for a fixed real data set.

6.5 Prospect and Future Work

A completely different approach to the one taken here would be to study the problem in a Bayesian framework. There are some works on this topic. Tipping and Bishop (1999) propose probabilistic principal components analysis which determines the principal components through maximum likelihood estimation of regression coefficients of a latent variable model that is closely related to factor analysis. Based on this, Wang (2012) builds a Bayesian principal components regression model with a dynamic component selection procedure. It has computational disadvantages since the MCMC adopted for estimation is time-consuming. Vidaurre et al. (2013) directly use the Bayesian method for the bilinear model of partial least squares regression. These works only introduce Bayesian statistics into principal components regression and partial least squares regression for

the estimation of regression coefficients, but they have not started to think about prediction uncertainty.

Section 2.2.3 shows it is not feasible to empirically estimate the slope and the intercept of the ordinary least squares prediction variance formula in order to give an approximate prediction variance for a fixed data set, although it is possible in theory. For principal components regression, Section 3.3.3 explores the relationship between the ordinary least squares type prediction mean squared error and the sample size, trying to find an empirical approximate prediction mean squared error for a fixed data set, but it fails. Inspired by these tries, it would be interesting for principal components regression and partial least squares regression, to adopt the Bayesian method to model an approximate linear relationship between prediction mean squared error and leverage.

References

Andersson, M. (2009). A comparison of nine PLS1 algorithms. *Journal of Chemometrics 23*(10), 518–529.

Baumann, K. and N. Stiefl (2004). Validation tools for variable subset regression. *Journal of Computer-Aided Molecular Design 18*(7-9), 549–562.

Belsey, D. A., E. Kuh, and R. E. Welsch (1980). *Regression diagnostics: identifying influential data and sources of collinearity*. New York: John Wiley.

Björkström, A. and R. Sundberg (1996). Continuum regression is not always continuous. *Journal of the Royal Statistical Society. Series B (Methodological) 58*(4), 703–710.

Copas, J. B. (1983). Regression, prediction and shrinkage. *Journal of the Royal Statistical Society. Series B (Methodological) 45*(3), 311–354.

De Jong, S. (1993). SIMPLS: an alternative approach to partial least squares regression. *Chemometrics and Intelligent Laboratory Systems 18*(3), 251–263.

De Vries, S. and C. J.F. Ter Braak (1995). Prediction error in partial least squares regression: a critique on the deviation used in the unscrambler. *Chemometrics and Intelligent Laboratory Systems 30*(2), 239–245.

Denham, M. C. (1997). Prediction intervals in partial least squares. *Journal of Chemometrics 11*(1), 39–52.

Dunn III, W., D. Scott, and W. Glen (1989). Principal components analysis and partial least squares regression. *Tetrahedron Computer Methodology 2*(6), 349–376.

Efron, B. (1982). *The Jackknife, the Bootstrap, and Other Resampling Plans*. Society for Industrial and Applied Mathematics.

Efron, B. and R. J. Tibshirani (1994). *An Introduction to the Bootstrap*. Chapman and Hall/CRC.

Faber, N. K. M. (2002). Uncertainty estimation for multivariate regression coefficients. *Chemometrics and Intelligent Laboratory Systems 64*(2), 169–179.

Faber, N. K. M. and R. Bro (2002). Standard error of prediction for multiway PLS: 1. background and a simulation study. *Chemometrics and Intelligent Laboratory Systems 61*(12), 133–149.

Faber, N. K. M. and B. R. Kowalski (1996). Prediction error in least squares regression: Further critique on the deviation used in the unscrambler. *Chemometrics and Intelligent Laboratory Systems 34*(2), 283–292.

Faber, N. K. M. and B. R. Kowalski (1997). Propagation of measurement errors for the validation of predictions obtained by principal component regression and partial least squares. *Journal of Chemometrics 11*(3), 181–238.

Fearn, T. and A. Davies (2003). Locally-biased regression. *Journal of Near Infrared Spectroscopy 11*(1), 467.

Fernández Pierna, J., L. Jin, F. Wahl, N. K. M. Faber, and D. Massart (2003). Estimation of partial least squares regression prediction uncertainty when the reference values carry a sizeable measurement error. *Chemometrics and Intelligent Laboratory Systems 65*(2), 281–291.

Filzmoser, P., B. Liebmann, and K. Varmuza (2009). Repeated double cross validation. *Journal of Chemometrics 23*(4), 160–171.

Frank, l. E. and J. H. Friedman (1993). A statistical view of some chemometrics regression tools. *Technometrics 35*(2), 109–135.

Geladi, P. and B. R. Kowalski (1986). Partial least-squares regression: a tutorial. *Analytica Chimica Acta 185*, 1–17.

Helland, I. S. (1988). On the structure of partial least squares regression. *Communications in Statistics - Simulation and Computation 17*(2), 581–607.

Helland, I. S. (1990). Partial least squares regression and statistical models. *Scandinavian Journal of Statistics 17*(2), 97–114.

Höskuldsson, A. (1988). PLS regression methods. *Journal of Chemometrics 2*(3), 211–228.

Høy, M., K. Steen, and H. Martens (1998). Review of partial least squares regression prediction error in unscrambler. *Chemometrics and Intelligent Laboratory Systems 44*(1-2), 123–133.

Jolliffe, I. T. (2002). *Principal Component Analysis* (2nd ed.). Springer.

Magnus, J. R. and H. Neudecker (1979). The commutation matrix: Some properties and applications. *The Annals of Statistics 7*(2), 381–394.

Manne, R. (1987). Analysis of two partial-least-squares algorithms for multivariate calibration. *Chemometrics and Intelligent Laboratory Systems 2*(13), 187–197.

Martens, H. and M. Martens (2000). Modified jack-knife estimation of parameter uncertainty in bilinear modelling by partial least squares regression (PLSR). *Food Quality and Preference 11*(12), 5–16.

Martens, H. and T. Næs (1991). *Multivariate Calibration* (New ed.). Wiley-Blackwell.

Olivieri, A. C., N. K. M. Faber, J. Ferr, R. Boqu, J. H. Kalivas, and H. Mark (2006). Uncertainty estimation and figures of merit for multivariate calibration (IUPAC technical report). *Pure and Applied Chemistry 78*(3), 633–661.

Phatak, A., P. Reilly, and A. Penlidis (2002). The asymptotic variance of the univariate PLS estimator. *Linear Algebra and its Applications 354*(13), 245–253.

Romera, R. (2010). Prediction intervals in partial least squares regression via a new local linearization approach. *Chemometrics and Intelligent Laboratory Systems 103*(2), 122–128.

Rosipal, R. and N. Krämer (2006). Overview and recent advances in partial least squares. In C. Saunders, M. Grobelnik, S. Gunn, and J. Shawe-Taylor (Eds.), *Subspace, Latent Structure and Feature Selection*, Number 3940 in Lecture Notes in Computer Science, pp. 34–51. Springer Berlin Heidelberg.

Rubinstein, R. (1982). Generating random vectors uniformly distributed inside and on the surface of different regions. *European Journal of Operational Research 10*(2), 205–209.

Searle, S. R. (1997). *Linear models*. Wiley.

Serneels, S., P. Lemberge, and P. J. Van Espen (2004). Calculation of PLS prediction intervals using efficient recursive relations for the jacobian matrix. *Journal of Chemometrics 18*(2), 76–80.

Stoica, P. and T. Söderström (1998). Partial least squares: A first-order analysis. *Scandinavian Journal of Statistics 25*(1), 17–24.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society. Series B (Methodological) 36*(2), 111–147.

Stone, M. and R. J. Brooks (1990). Continuum regression: Cross-validated sequentially constructed prediction embracing ordinary least squares, partial least squares and principal components regression. *Journal of the Royal Statistical Society. Series B (Methodological) 52*(2), 237–269.

Tipping, M. E. and C. M. Bishop (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 61*(3), 611–622.

Van Der Voet, H. (1999). Pseudo-degrees of freedom for complex predictive models: the example of partial least squares. *Journal of Chemometrics 13*(3-4), 195–208.

Vidaurre, D., M. A. J. van Gerven, C. Bielza, P. Larraaga, and T. Heskes (2013). Bayesian sparse partial least squares. *Neural Computation 25*(12), 3318–3339.

Wang, L. (2012). Bayesian principal component regression with data-driven component selection. *Journal of Applied Statistics 39*(6), 1177–1189.

Wehrens, R. and W. E. Van Der Linden (1997). Bootstrapping principal component regression models. *Journal of Chemometrics 11*(2), 157–171.

Wold, H. (1966). Estimation of principal components and related models by iterative least squares. In *Multivariate Analysis* (Krishnaiaah , P. R. ed.)., pp. 391–420. New York: Academic Press.

Wold, H. (1973). Nonlinear iterative partial least squares (NIPALS) modeling: some current developments. In *Krishnaiah, P. R., ed. Multivariate Analysis II. Proc. Int. Symp. Multivariate Anal. held at Wright State University, Dayton, Ohio, June 1924, 1972*, pp. 383–407. New York: Academic Press.

Wold, S., H. Martens, and H. Wold (1983). The multivariate calibration problem in chemistry solved by the PLS method. In B. Kgstrm and A. Ruhe (Eds.), *Matrix Pencils*, Number 973 in Lecture Notes in Mathematics, pp. 286–293. Springer Berlin Heidelberg.

Wold, S., M. Sjöström, and L. Eriksson (2001). PLS-regression: a basic tool of chemometrics. *Chemometrics and Intelligent Laboratory Systems 58*(2), 109–130.

Xu, Q., Y. Liang, and Y. Du (2004). Monte carlo cross-validation for selecting a model and estimating the prediction error in multivariate calibration. *Journal of Chemometrics 18*(2), 112–120.

Ye, J. (1998). On measuring and correcting the effects of data mining and model selection. *Journal of the American Statistical Association 93*(441), 120–131.

Zhang, L. and S. Garcia-Munoz (2009). A comparison of different methods to estimate prediction uncertainty using partial least squares (PLS): a practitioner's perspective. *Chemometrics and Intelligent Laboratory Systems 97*(2), 152–158.